UNIVERSIDADE DE LISBOA
Faculdade de Ciências
Departamento de Informática

ANÁLISE E IMPLEMENTAÇÃO DE SISTEMAS IDS E IPS

Guilherme Filipe Zorego Rodrigues Morais

MESTRADO EM ENGENHARIA INFORMÁTICA
Arquitectura, Sistema e Redes de Computadores

2011

UNIVERSIDADE DE LISBOA
Faculdade de Ciências
Departamento de Informática

ANÁLISE E IMPLEMENTAÇÃO DE SISTEMAS IDS E IPS

Guilherme Filipe Zorego Rodrigues Morais

PROJECTO

Trabalho orientado pelo Prof. Doutora Maria Dulce Pedroso Domingos

e co-orientado por Dr. Pedro Miguel Gomes Silva Rosa.

MESTRADO EM ENGENHARIA INFORMÁTICA
Arquitectura, Sistemas e Redes de Computadores

2011

Agradecimentos

Em primeiro lugar gostaria de agradecer à Professora Dulce Domingos pela paciência e dedicação em rever os inúmeros "drafts" enviados.

Ao Co-Orientador Pedro Rosa pela oportunidade de elaborar este projecto e aprofundar os meus conhecimentos.

À minha namorada pela dedicação e apoio incondicional.

Ao Paulo Bastos pela enorme ajuda técnica dada.

Aos meus colegas, Jorge Sá Pereira e Francisco Estanqueiro pelo apoio e bons momentos passados no Centro de Informática

Aos meus pais e à minha irmã pelo apoio naqueles momentos menos bons.

E por fim a todos os meus amigos que nunca deixaram de acreditar em mim.

Resumo

Com o aumento dos serviços *online* fornecidos pelo Centro de Informática da Faculdade de Ciências e com o crescente número de utilizadores desses serviços, o risco de ataques às infra-estruturas aumenta.

A infra-estrutura da rede da Faculdade de Ciências inclui Firewalls e VPNs. No entanto não é suficiente para proteger contra todo o tipo de ataques.

Com este projecto pretendeu-se estudar e implementar um sistema de detecção de intrusões de modo a contemplar ataques que de outra forma não seria possível detectar com métodos tradicionais.

O principal objectivo do projecto consistiu na criação de um sistema de detecção de intrusões para a rede da Faculdade de Ciências, para em conjunto com os sistemas já existentes melhorar a segurança.

Foi recolhida informação sobre as diferentes soluções disponíveis e foram analisadas e testadas algumas das soluções que poderiam vir a servir para melhorar o sistema de segurança.

Para perceber o meio em que o sistema de detecção de intrusões iria estar integrado foi feita uma análise à rede e infra-estrutura.

Depois de recolhidas os dados sobre as soluções disponíveis e o ambiente onde o sistema ia estar inserido foram enumerados os diferentes requisitos que a solução deve preencher. Como resultado da análise desses requisitos a solução recaiu numa solução híbrida, uma combinação de duas ferramentas o Ossec e o Snort.

Foram implementadas as ferramentas e posteriormente houve uma fase de testes funcionais e de impacto.

Com a implementação e configuração dos sistemas é agora possível aos técnicos do Centro de informática a monitorização das redes e infra-estruturas de forma a detectarem possíveis ataques à rede da faculdade.

Abstract

With the increase of the online services provided by the Computing Centre of the Faculty of Sciences and the increasing number of users, the risk of attacks on the infrastructure increases.

The infrastructure network of the Faculty of Science includes firewalls and VPNs. However it is not enough to protect against all types of attacks.

With this project we intended to study and implement an intrusion detection system in order to detect attacks that otherwise would not be possible to detect with traditional methods.

The project's main objective was the deployment of an intrusion detection system for the network of the Faculty of Sciences to improve security, in conjunction with existing systems.

Information was collected on the different solutions available and were analyzed and tested different solutions that could eventually serve to improve the security system.

To understand the environment in which the intrusion detection system would be integrated into an analysis was made to the network and infrastructure.

Once collected the data about the available solutions and the environment where the system would be inserted, the different requirements that the solution must satisfy were listed. As a result of the analysis of these requirements the solution was based on a hybrid solution, a combination of two tools OSSEC and Snort.

After the tools implementation there were functional and impact testing.

With the implementation and configuration of systems is now possible to technicians of the Center to monitoring computer networks and infrastructure in order to detect possible attacks.

Conteúdo

Capítulo 1 Introdução 1

1.1 Motivação 1

1.2 Objectivos 1

1.3 Instituição de Acolhimento 2

1.4 Estrutura do documento 3

1.5 Planeamento 4

Capítulo 2 Segurança 5

2.1 Conceitos 5

2.2 Ataques 6

2.2.1 Métodos de Ataque 7

2.2.2 Exemplos de ataques 10

2.3 Actualidade 13

Capítulo 3 Sistema de Detecção de Intrusões 17

3.1 Firewall vs IDS 17

3.2 Tipos de IDS 18

3.2.1 Fonte de eventos 18

3.2.2 Método de Detecção 19

3.2.3 Frequência de utilização 19

3.2.4 Fiabilidade 20

3.2.5 Reactividade 20

3.3 Honey Pot 20

3.4 Principais características de um IDS 21

3.5 Limitações de um sistema IDS 22

3.6 Ferramentas IDS 23

3.6.1 OSSEC 23

3.6.2 Snort 23

3.6.3 AntiSniff 23

3.6.4 TripWire 24
3.6.5 Valhala 24
3.6.6 Tabela comparativa 24
Capítulo 4 Rede FCUL 25
4.1 Estrutura da rede 25
4.2 Estrutura Física 26
4.3 Serviços de Rede 26
4.3.1 Domain Controller 26
4.3.2 DHCP 27
4.3.3 DNS Interno 27
4.3.4 DNS Externo 28
4.3.5 Servidor de correio electrónico 28
4.4 Serviços Web 29
4.4.1 Gestão FCUL 29
4.4.2 Moodle FCUL 29
4.4.1 Webmail FCUL 29
4.4.2 Single Sign On FCUL 30
4.5 Segurança 30
4.5.1 Sistemas de Segurança 31
4.5.2 Políticas e procedimentos 31
4.5.3 Segurança Física 32
Capítulo 5 Sistema de Detecção de Intrusões FCUL 33
5.1 Requisitos da solução IDS a implementar 33
5.2 Arquitectura OSSEC 34
5.2.1 Processos Internos 35
5.2.2 Fluxo dos ficheiros de log 36
5.2.3 Análise dos logs 36
5.2.4 Segurança interna 40
5.2.5 Comunicação 40

5.2.6 Active Response 41
5.2.7 Verificação de Integridade 43
5.3 Arquitectura SNORT 44
5.3.1 Componentes SNORT 44
Capítulo 6 Implementação e Testes 47
6.1 Implementação do Servidor IDS 47
6.1.1 Instalação 47
6.1.2 Configuração OSSEC 48
6.1.3 Configuração Snort 49
6.2 Configuração dos Agentes 50
6.3 Configuração Front-End 50
6.4 Testes 51
6.4.1 Ferramentas de Teste 51
6.4.2 Testes Funcionais 52
6.4.3 Testes de Impacto 57
6.4.4 Tabela de resultados e Conclusão 62
Capítulo 7 Conclusão e Trabalho Futuro 65
7.1 Dificuldades encontradas 66
7.2 Trabalhos Futuros 66
Capítulo 8 Referências 67
Capítulo 9 Glossário 69
Capítulo 10 Abreviaturas 71

Lista de Figuras

Figura 1.1	:	Mapa de Gantt	4
Figura 2.1	:	Actividade maliciosa por país	14
Figura 2.2	:	Ataques "Web-based" mais comuns	14
Figura 2.3	:	Bens e serviços à venda no mercado paralelo	15
Figura 2.4	:	Código malicioso mais comum	15
Figura 2.5	:	Sectores mais afectados por ataques de phishing	16
Figura 4.1	:	Rede FCUL	25
Figura 4.2	:	Diagrama do SSO da FCUL	30
Figura 5.1	:	Fluxo dos ficheiros de log	36
Figura 5.2	:	Fluxo interno dos ficheiros de log	36
Figura 5.3	:	Fluxo dos ficheiros de log	41
Figura 5.4	:	Fluxo de processamento do Snort	44
Figura 6.1	:	Gráfico de utilização de processador, sem agente	58
Figura 6.2	:	Gráfico de utilização de processador, com agente	59
Figura 6.3	:	Gráfico de consumo de memória, sem agente	59
Figura 6.4	:	Gráfico de consumo de memória, com agente	60
Figura 6.5	:	Gráfico de utilização de processador, sem agente	60
Figura 6.6	:	Gráfico de utilização de processador, com agente	61
Figura 6.7	:	Gráfico de consumo de memória, sem agente	61
Figura 6.8	:	Gráfico de consumo de memória, com agente	61

Lista de Tabelas

Tabela 3.1	:	Tabela Comparativa entre os vários IDS	24
Tabela 5.1	:	Parâmetros da função de Pre-decoding	37
Tabela 5.2	:	Parâmetros da função de Decoding	38
Tabela 6.1	:	Resultados dos testes funcionais	62
Tabela 6.2	:	Resultado dos testes de impacto no Windows XP sp2	62
Tabela 6.3	:	Resultado dos testes de impacto no Ubuntu 9.10	63

Capítulo 1 Introdução

Este projecto pretende estudar e implementar um sistema de IDS (*Intrusion Detection System*)/ IPS (*Intrusion Protection System*) nas redes geridas pelo Centro de Informática da Faculdade de Ciências da Universidade de Lisboa de forma a aumentar a sua segurança. Este projecto encontra-se no âmbito da cadeira de Projecto de Engenharia Informática do mestrado em Engenharia Informática.

1.1 Motivação

Uma das principais motivações para a realização deste projecto está relacionada com o facto de este vir a melhorar a segurança e disponibilidade dos serviços e recursos disponibilizados pela Faculdade de Ciências.

Estes serviços têm crescido ao longo dos anos, tendo-se tornado mais complexos e mais abrangentes. Estes vão desde o pagamento das propinas ao lançamento de notas, passando por outros serviços como servidores de Correio Electrónico, *DNS*, *DHCP* e alojamento de páginas e áreas pessoais.

Muitos destes serviços podem ser acedidos internamente ou externamente, logo é necessário garantir a sua segurança.

Apesar do Centro de Informática já possuir uma solução baseada em *firewalls*, é necessário implementar um sistema que tenha a capacidade de lançar um alerta em tempo real e responder de forma reactiva de acordo com um conjunto de regras previamente definidas.

1.2 Objectivos

O principal objectivo é analisar a solução IDS/IPS que melhor se adequa à infra-estrutura de redes e servidores geridas pelo centro de informática, de forma a "defendê-las" dos ataques mais comuns.

Com a instalação de uma solução IDS/IPS vai ser possível uma monitorização da rede eficiente e desta forma desenvolver uma resposta a possíveis ataques.

Para tal é necessário fazer um levantamento das características das diversas redes e infra-estruturas geridas pelo Centro de Informática e analisar os benefícios e consequências de uma solução baseada em *open-source*.

A opção *open-source* foi tomada visto que não tem custos directos para a sua implementação e sendo *open-source* existe a possibilidade de modificar o software para que este se adeque às nossas necessidades.

1.3 Instituição de Acolhimento

A Faculdade de Ciências da Universidade de Lisboa – FCUL, é uma das unidades orgânicas que integram a Universidade de Lisboa. A FCUL ocupa onze edifícios com uma área total de 77 492m2, no campus do Campo Grande e tinha em 2008/2009, 5061 alunos distribuídos pelos vários ciclos de ensino e 634 funcionários, entre os quais 419 docentes.

O Centro de Informática é a Unidade Orgânica da FCUL responsável pela gestão da rede Internet, bem como dos diversos serviços a ela associados, nomeadamente o correio electrónico e páginas Web. A manutenção de alguns sistemas de informação, tais como as bases de dados relativas à gestão de utilizadores (docentes, funcionários e alunos) e correspondentes (URL) páginas amarelas, está também a cargo do Centro.

O Centro de Informática tem ainda a seu cargo o apoio técnico a todos os funcionários da Faculdade, através de um serviço de *Help Desk*. Contudo, a prestação de apoio especializado é realizada no âmbito de acordos de colaboração com as várias unidades orgânicas da FCUL.

A equipa do Centro de Informática encontra-se actualmente dividida em 4 áreas principais:

- **Administração de Redes**: Responsável pela instalação, manutenção e configuração de todo o equipamento de rede presente na Faculdade (*Routers, Switchs, Firewalls, Acess Points*) e controle dos acessos à rede e monitorização de tráfego.

- **Administração de Sistemas e Bases de Dados**: Responsável pela instalação, manutenção e configuração de todos os Servidores presentes na Faculdade (*Mail, Active Directory , DHCP, DNS, VmWare, Moodle*...). Administração e Manutenção das Bases de Dados dos Sistemas.

  Suporte à criação e gestão de utilizadores da faculdade (Alunos, Professores, Funcionários, Estagiários, Investigadores...).

- **Área de Desenvolvimento**: Responsável pela criação e desenvolvimento de novos Sites, Aplicações e Plataformas para uso interno do Centro de Informática ou para uso específico de algum Departamento ou órgão pertencente à Faculdade.

- **Área de Suporte/Help Desk**: Responsável pelo fornecimento de suporte directo aos Professores e Funcionários nos seus problemas diários (instalação de impressoras, problemas de acesso à internet, configuração de contas de correio electrónico...). Suporte e manutenção da rede (Activação e desactivação de tomadas). Instalação e manutenção de equipamentos (instalação e configuração de computadores e impressoras, formatação de computadores, remoção de vírus e *spywares*, criação de backups, instalação de novos sistemas operativos e software).

1.4 Estrutura do documento

O documento está organizado da seguinte forma:

- **Capítulo 2** - Segurança - Neste capítulo são explicados alguns conceitos sobre a segurança, ataques e é ainda apresentada uma breve análise ao relatório da Symantec.
- **Capítulo 3** - IDS - Neste capítulo é feita uma introdução aos sistemas de IDS e uma análise das soluções mais importantes disponíveis.
- **Capítulo 4** - Rede FCUL - Neste capítulo é feita uma análise à infra-estrutura e serviços geridos pelo Centro de Informática e às políticas de segurança na FCUL.
- **Capítulo 5** - Sistema de detecção de intrusões na FCUL - Neste capítulo são apresentados os requisitos necessários à implementação de um sistema IDS e é apresentada a solução escolhida e explicado o seu funcionamento.
- **Capítulo 6** - Implementação e Testes - Neste capítulo é explicada a implementação da solução e são apresentados os testes feitos.
- **Capítulo 7** - Conclusão e Trabalhos Futuros - Neste capítulo são apresentadas as conclusões finais do projecto, as dificuldades encontradas e o trabalho futuro

1.5 Planeamento

Esta secção apresenta o planeamento previsto para o projecto.

- *Janeiro 2010 e Fevereiro 2010:*

  - Estudo do funcionamento dos sistemas de IDS/IPS.

  - Levantamento das características das diversas sub-redes da FCUL e classificação da sua importância.

  -Elaboração do relatório Preliminar.

- *Março 2010:*

  -Análise das soluções IDS existentes elaborando quadros comparativos adequados.

  -Elaboração do índice e introdução dos capítulos do Relatório Final

- Abril e Maio 2010:

  - Estudo sobre o impacto (indesejado) das medidas a concretizar para implementação de sistemas IDS/IPS.

  - Estudo sobre onde concretizar essas medidas.

  - Analisar os benefícios esperados em termos de aumento de segurança e monitorização.

  - Complementar o Relatório com os dados obtidos nesta fase.

- *Junho 2010:*

  - Implementação e validação da solução considerada mais adequada.

  - Testes funcionais e de impacto da solução.

  - Complementar o Relatório com os dados obtidos nesta fase.

- *Julho e Agosto 2010:*

  - Testes da implementação com elaboração dos refinamentos considerados adequados.

  - Elaboração da Documentação apropriada para o CI dar continuidade à utilização das ferramentas.

  -Finalização do Relatório final.

	Nome	Início	Término
1	Estudo do funcionamento dos sistemas IDS/IPS.	04-01-2010 8:00	26-02-2010 17:00
2	Levantamento das características das diversas sub-redes da FCUL e classificação da sua importância.	04-01-2010 8:00	26-02-2010 17:00
3	Elaboração do relatório preliminar.	04-01-2010 8:00	26-02-2010 17:00
4	Análise das soluções IDS existentes elaborando quadros comparativos adequados.	01-03-2010 8:00	31-03-2010 17:00
5	Elaboração do índice e introdução dos capítulos do Relatório Final	01-03-2010 8:00	31-03-2010 17:00
6	Estudo sobre o impacto (indesejado) das medidas a concretizar para implementação de sistemas IDS/IPS.	01-04-2010 8:00	20-05-2010 17:00
7	Estudo sobre onde concretizar essas medidas.	01-04-2010 8:00	28-05-2010 17:00
8	Analisar os benefícios esperados em termos de aumento de segurança e monitorização	01-04-2010 8:00	28-05-2010 17:00
9	Complementar o Relatório com os dados obtidos nesta fase.	01-04-2010 8:00	28-05-2010 17:00
10	Implementação e validação da solução consideradas mais adequada.	01-06-2010 8:00	30-06-2010 17:00
11	Testes funcionais e de impacto da solução.	01-06-2010 8:00	30-06-2010 17:00
12	Complementar o Relatório com os dados obtidos nesta fase.	01-06-2010 8:00	30-06-2010 17:00
13	Testes da implementação com elaboração dos refinamentos considerados adequados.	01-07-2010 8:00	20-08-2010 17:00
14	Elaboração da Documentação apropriada para o CI dar continuidade à utilização das ferramentas.	01-07-2010 8:00	20-08-2010 17:00
15	Finalização do Relatório final.	01-07-2010 8:00	20-08-2010 17:00

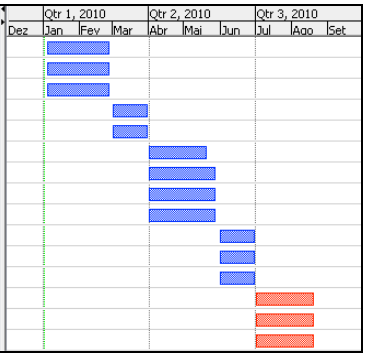

Figura 1.1: Mapa de Gantt.

Capítulo 2 Segurança

Neste capítulo são apresentados alguns conceitos introdutórios de segurança e é feita uma análise do relatório de segurança publicado pela *Synmatec*.

2.1 Conceitos

A Segurança dos serviços e informação numa instituição ou empresa refere-se à protecção dos dados e serviços existentes que muitas vezes são o negócio da empresa.

A falta de segurança está associada a deficiências técnicas e à atitude das pessoas.

A insegurança é uma das consequências directa das acções das pessoas que precisam de ser educadas sobre quão sérios são os seus actos/obrigações.

Segundo William Stallings os serviços de segurança devem conter as seguintes propriedades[1]:

- **Autenticidade** - Medida que assegura que a entidade que está a comunicar é quem ela realmente diz ser
- **Controle de acesso** - Medida que previne o acesso aos recursos de pessoas não autorizadas.
- **Confidencialidade** - Medida em que um serviço/informação está protegido contra o acesso de leitura de intrusos.
- **Integridade** - Medida em que o serviço/informação está protegido contra a modificação/deterioração por intrusos.
- **Disponibilidade** - Medida em que um serviço/informação está protegido contra a recusa de provisão/acesso provocada por intrusos.
- **Não repudiação** - Medida em que deve ser possível provar a origem dos dados.

Alguns dos mecanismos para garantir estas propriedades são:

- Cifra de modo a garantir a confidencialidade dos dados.
- Autenticação prévia dos intervenientes de modo a garantir a autenticidade dos dados.
- A criação de *hash* para garantir a integridade.
- Assinaturas digitais dos documentos de modo a garantir a não repudiação.

Segundo André Zúquete [2] uma intrusão é "qualquer conjunto de acções com o intuito de comprometer a integridade, a confidencialidade ou a disponibilidade de um recurso."

Uma intrusão é o resultado de um ou mais ataques podendo estes provocar alterações nos dados guardados no sistema.

Uma intrusão tem duas causas:

- **Vulnerabilidade** - É uma característica do sistema que o torna susceptível a certos ataques, uma fraqueza no sistema que pode ser explorada com intenções maliciosas.
- **Ataque** - É um conjunto de acções que visam explorar as vulnerabilidades e que permitem uma acção ilícita.

Uma intrusão é o resultado de um ataque bem sucedido a uma vulnerabilidade.

2.2 Ataques

Um ataque é uma acção hostil e deliberada com a intenção de quebrar a segurança de um sistema ou infra-estrutura, podendo ser manual ou automática.

Esta acção corrompe, degrada ou destrói a informação contida nos computadores e na rede de computadores ou até os próprios computadores e redes.

Segundo Paulo Veríssimo[4] existem vários factores que motivam alguém a tentar invadir um sistema, alguns deles são a curiosidade; colecção de troféus; acesso gratuito a recursos computacionais e de comunicações; para servir de ponte para outras

máquinas num sistema distribuído; para sabotagem ou danificar um sistema, por razões criminais, mercenárias ou politicas; para o roubo de informação, para uso próprio ou para venda, como software, ou segredos comerciais ou industriais.

2.2.1 Métodos de Ataque

Segundo Paulo Sousa[3] os Ataques podem ser classificados de duas formas:

- **Manual -** Num ataque manual, o alvo é escolhido cuidadosamente e o objectivo é bem definido antes de ser escolhida a técnica de ataque.
  Este tipo de ataque é composto por 8 fases:

  -Footprinting- Onde é feita a recolha inicial sobre os potenciais alvos (endereços de IP, protocolos, sistemas ligados à Internet);
  -Fingerprinting (Varrimento) - É feita uma procura por sistemas acessíveis, portos abertos (*port scanning*), descoberta do software usado e as suas versões;
  -Enumeração - Acção mais intrusiva que visa recolher informação sobre os recursos/partilhas de rede, nomes de utilizadores/grupos e aplicações.
  - *Identificação de Vulnerabilidades* - Manualmente ou automaticamente (Nessus, OpenVAS);
  - *Exploração de vulnerabilidade(s)/Ataque*;
  - *Elevação de privilégios* (explorando outras vulnerabilidades);
  - *Instalação de rootkit* (backdoor,zombie/bot,...);
  - *Remoção de Pistas* (apagar/editar registos de log);

- **Automáticos** - Os ataques automáticos são aqueles que não necessitam de intervenção humana para serem efectivados, podendo ser activados através de *scripts* ou software específico.
  Este tipo de ataque é composto por 5 fases:

  -***Footprinting***- Onde é feita a recolha inicial sobre os potenciais alvos (endereços de IP, protocolos, sistemas ligados à Internet);

-***Fingerprinting*** (Varrimento) - É feita uma procura por sistemas acessíveis, portos abertos (*port scanning*), descoberta do software usado e as suas versões;

-**Enumeração** - Acção mais intrusiva que visa recolher informação sobre os recursos/partilhas de rede, nomes de utilizadores/grupos e aplicações;

- **Ataque**;

- **Instalação de *rootkit*** (backdoor,zombie/bot,...);

De seguida são explicados algumas formas de ataques automatizados, tais como vírus, *worms*, cavalos de tróia e *exploits*:

- **Vírus** - São programas maliciosos que infectam os sistemas modificando e corrompendo os ficheiros. A contaminação dos sistemas ocorre habitualmente pela acção do utilizador que executa um determinado programa que à partida pode parecer inofensivo.

  A sua detecção é feita pelos Antivírus que contêm uma base de dados de assinaturas e pedaços de código de vírus podendo desta forma reconhece-los e em alguns casos até corrigir os ficheiros infectados.

  São classificados de ataques automatizados visto que a sua capacidade de disseminação depende do sistema atacado e não do atacante.

- **Worms** – São programas maliciosos semelhantes a um vírus, mas mais complexos. Um *worm* não necessita de um programa hospedeiro para se propagar, pode-se propagar através de mensagens de correio electrónico, ligações de rede ou até dissimulados em ficheiros aparentemente inofensivos. Os *worms* são detectados pelo mesmo software que detecta vírus. Assim como os vírus, são considerados ataques automatizados, visto que a sua capacidade de disseminação é independente da acção do atacante.

- **Cavalo de Tróia -** Cavalo de Tróia ou Trojan Horse é um programa malicioso que vem dissimulado dentro de um programa que actua de forma legítima.

Os *Trojan Horses* podem ser usados para executar várias acções no computador infectado, tais como:

- Criar *backdoors* para que o atacante tenha total acesso ao computador e ao seu conteúdo.
- Após estarem alojados podem transmitir todo um conjunto de informação sensível como passwords e números de cartões de crédito.
- Esgotar o espaço no disco criando ficheiros ao acaso.
- Bloquear o computador do utilizador.
- Apagar ou modificar ficheiros críticos do sistema.
- Enviar para o atacante *screen-shots* do ecrã do utilizador.
- Enviar informação das teclas que o utilizador está a digitar (*Keystroke logging*).
- Alojar-se na máquina e mantêrem-se inactivos até que recebam ordens para actuarem como máquinas de *spam* ou para ataques de *DOS*.

  É possível que este tipo de software se replique através de um programa de *instant-messaging* ou através de correio electrónico.

- **Exploit** - Um exploit é um programa que se aproveita de uma vulnerabilidade de um sistema para desta forma tentar obter o controlo do sistema. Habitualmente essas vulnerabilidades são reportadas pela entidade criadora do software que consequentemente desenvolve um *patch* para a correcção, porém estas correcções nem sempre são instaladas.

  É considerado um ataque automático visto que o atacante apenas executa a sua ferramenta, sendo que as invasões acontecem de forma automática.

2.2.2 Exemplos de ataques

Existem vários tipos de ataques que diferem pela forma como são feitos e pelo objectivo.

Alguns do ataques mais comuns são:

- ***BackDoor*** - Uma *BackDoor* é um método que permite o acesso a um sistema ultrapassando as verificações de segurança. Muitas das Backdoor's existentes são criadas pelos criadores do sistema de forma a pouparem tempo e esforço quando estão a detectar problemas. As Backdoors são uma ameaça aos sistemas e devem ser evitadas ao máximo. Se uma Backdoor é descoberta por um atacante pode ser usada para invadir o sistema.
- ***Password Guessing Attack*** - Este tipo de ataques ocorre quando um utilizador não autorizado, tenta repetidamente aceder a um computador ou rede tentando adivinhar nomes de utilizadores e palavra-chave.

  Este tipo de ataque divide-se em duas categorias:

  - **Brute Force Attack** - Neste tipo de ataques o atacante tenta adivinhar a palavra-chave de um utilizador gerando todas as combinações de palavras-chave possíveis.
  - **Dictionary Attack** - Neste tipo de ataque, o atacante usa uma lista de palavras comuns para adivinhar a palavra-chave. Esta lista de palavras pode ser usada com letras maiúsculas e minúsculas.

- **DOS** – DOS (*denial-of-service*) ou ataques de negação de serviço, são tipos de ataques que pretendem esgotar os recursos da máquina ou torná-la inacessível a um determinado utilizador ou máquina.

Pode ser efectuado por uma só máquina, mas normalmente são usadas múltiplas máquinas, denominado de DDOS (*Distributed Denial of Service*) de forma a tornar o ataque mais eficiente. Essas máquinas são máquinas *zombie*, previamente "capturadas" pelo Atacante através de vírus e *worms*.

Alguns dos ataques DOS são:

- **SYN Attack** - É um tipo de DoS , também conhecido como SYN flooding, em que o atacante envia múltiplos pacotes SYN para um computador alvo. Para cada pacote SYN recebido pelo computador alvo este responde com um pacote de acknowledgment (SYN-ACK) para o IP de origem. Não obtendo resposta por parte da origem tenta reenviar o pacote SYN-ACK. Este tipo de acção deixa as portas TCP meio abertas à espera de completar a ligação. Quando este tipo de ataque é executado repetidamente a máquina alvo eventualmente ficará sem recursos para responder a todos os pedidos de ligação, desta forma negando o acesso a utilizadores legítimos.
- **Ping Attack** - Neste tipo de ataque DoS, o atacante envia repetidamente pacotes ICMP com um tamanho acima do normal para o computador alvo. Este tipo de ataques estão direccionados para pilhas TCP que não conseguem gerir pacotes ICMP. Este ataque inunda o servidor de destino com pacotes falsos.
- **Flood Attack** - Neste tipo de ataque DoS o atacante envia permanentemente enormes quantidades de tráfego. Tráfego esse superior àquele que o computador alvo consegue processar.
- **Teardrop Attack** - Neste ataque são enviados pacotes corruptos para o computador alvo, usando o algoritmo de fragmentação de pacotes IP. Como resultado deste ataque o computador da vítima poderá deixar de responder.
- **Smurf Attack** - Neste ataque, o atacante envia um grande número de ICMP *echo requests* para o IP de *broadcast* usando um IP de origem falsa. Estes pedidos vão parecer que tiveram origem na rede da vítima, fazendo com que todos os computadores no mesmo domínio da vítima respondam à vítima. Como resultado o computador da vítima será inundado de respostas.

- **Spoofing** – O Spoofing é um ataque que é caracterizado pela criação de pacotes TCP/IP usando o endereço IP de outro utilizador. Os routers usam o IP de destino de forma a poderem encaminhar os seus pacotes ao longo da rede, mas ignoram o endereço IP de origem. Esse endereço é apenas usado pela máquina de destino quando responde.

  Alguns exemplos de ataques de spoofing são:

  - **Man-in-the-Middle Attack** - Este tipo de ataque ocorre quando um atacante consegue inserir um software intermediário ou programa entre dois hosts que estão a comunicar. Esta posição intermediária permite ao atacante ouvir e modificar os pacotes de comunicação que passam entre os dois hosts. O software intercepta os pacotes de comunicação e posteriormente envia a informação ao host de recepção. O receptor responde ao software, presumindo que é um cliente legítimo.
  - **Flooding Attack** - Este tipo de ataque ocorre quando um atacante consegue inserir um software intermediário ou programa entre dois hosts que estão a comunicar. Esta posição intermediária permite ao atacante ouvir e modificar os pacotes de comunicação que passam entre os dois hosts.

2.3 Actualidade

Para se perceber como se encontra o panorama global da segurança, a *Symantec Global Intelligence Network*, uma empresa mundial na área de segurança, tem ao seu dispor:

- Mais 240.000 sensores a monitorizar a actividade das redes e sistemas em mais de 200 países;
- Obtém dados sobre código malicioso junto de mais de 133 milhões de máquinas divididos entre clientes, servidores e gateways;
- Mantém ainda uma rede de *honeypots* em todo o mundo que recolhe informação, sobre ameaças e ataques ainda não reportados;
- Mantém uma das maiores bases de dados de vulnerabilidades do mundo, detalhando mais de 13.000 vulnerabilidades;
- Mais de 8 milhares de milhões de mensagens de email, assim como mais de 1 milhar de milhões de pedidos web, são processados todos os dias através de 16 *data centers*;
- Reúne informação sobre *phishing* através de uma extensa comunidade de empresas anti-fraude, vendedores de soluções de segurança e mais de 50 milhões de consumidores;

Após a análise do relatório de segurança da *Symantec Intelligence Quarterly* (Abril-Junho 2010), podem ser tiradas algumas ilações. [5]

- Como mostra a figura 2.1 os Estados Unidos da América são o país onde teve origem a grande maioria dos ataques (21% neste trimestre).

Rank	Country/Region	Percentage	Malicious Code Rank	Spam Zombies Rank	Phishing Website Hosts Rank	Bots Rank	Attack Origin Rank
1	United States	21%	1	5	1	1	1
2	India	6%	2	1	20	20	9
3	Germany	6%	21	4	2	2	7
4	China	5%	3	35	8	6	2
5	Brazil	5%	5	2	10	5	5
6	Italy	4%	15	8	13	4	6
7	United Kingdom	4%	10	9	3	8	3
8	Taiwan	3%	22	13	14	3	10
9	Russia	3%	12	7	7	15	4
10	France	3%	19	18	6	11	8

Figura 2.1: Actividade maliciosa por país

- A figura 2.2 mostra que o ataque "Web-based" mais comum no trimestre esteve relacionado com PDFs maliciosos, sendo estes 36% do total dos ataques "Web-Based"

Rank	Attack	Percentage
1	PDF Suspicious File Download	36%
2	Microsoft® Internet Explorer® ADODB.Stream Object File Installation Weakness	33%
3	C6 Messenger ActiveX File Overwrite	7%
4	Microsoft Internet Explorer DHTML CreateControlRange Code Executable	5%
5	Adobe® SWF Remote Code Execution	5%
6	Embed Tag NPDSPlay DLL Buffer Overflow	3%
7	Microsoft Internet Explorer WPAD Spoofing	2%
8	Microsoft Internet Explorer Popup Window Address Bar Spoofing Weakness	1%
9	Microsoft Internet Explorer CreateTextRange Remote Code Execution	1%
10	Microsoft Internet Explorer Malformed IFRAME/EMBED Buffer Overflow	1%

Figura 2.2: Ataques "Web-based" mais comuns

- A figura 2.3 mostra que os dados de cartões de crédito são o item mais anunciado para venda no mercado paralelo, que corresponde a 28% de todos os bens e serviços anunciados.

Rank	Item	Percentage	Range of Prices
1	Credit cards	28%	$1 - $30
2	Bank accounts	24%	$10 - $125
3	Email accounts	8%	$5 - $12
4	Email addresses	5%	$5 - $10 per MB
5	Credit card dumps	4%	No specified prices
6	R57 & C99 shells	3%	$2 - $5
7	Full identity	3%	$3 - $20
8	Mailers	3%	$1 - $5
9	Attack toolkits	3%	$5 - $20 or $120 per month
10	Cash-out services	2%	$200 - 100 or 50% - 70%

Figura 2.3: Bens e serviços à venda no mercado paralelo

- Na figura 2.4 podemos ver que o código malicioso mais comum durante este trimestre foi o vírus Sality.AE.

Rank	Sample	Type	Infection Vector(s)	Impact
1	Sality.AE	Virus, worm	Executables	Removes security applications and services, downloads and installs additional threats
2	Mabezat.B	Worm, virus	SMTP, CIFS, removable drives	Encrypts and infects files
3	Downadup.B	Worm	Direct network connections, CIFS	Disables security applications and Windows Update, downloads and installs additional threats
4	Virut.CF	Virus	Executables	Infects files, downloads and installs additional threats
5	SillyFDC	Worm	Mapped, removable drives	Downloads and installs additional threats
6	Almanahe	Worm, virus	CIFS, mapped and removable drives	Infects executable files, ends security related processes and installs additional threats
7	Gammima.AG	Worm, virus	Removable drives	Steals online game account credentials
8	FakeAV	Trojan	N/A	Displays false security alerts and lowers security settings
9	Gampass	Trojan	N/A	Steals online game account credentials
10	Chir.B@mm	Virus, worm	SMTP	Infects executable and HTML files

Figura 2.4: Código malicioso mais comum

- Na figura 2.5 podemos ver que sector mais afectado por ataques de *phishing* foi sector financeiro que corresponde a 73% de todos os ataques de *phishing.*

Rank	Sector	Percentage
1	Financial	73%
2	ISP	10%
3	Retail	5%
4	Insurance	3%
5	Internet community	2%
6	Government	2%
7	Telecom	2%
8	Computer hardware	2%
9	Online gaming	<1%
10	Computer consulting	<1%

Figura 2.5: Sectores mais afectados por ataques de *phishing*

Durante este semestre a Symantec criou ainda 457.641 novas assinaturas de código malicioso e observou 12.7 biliões de mensagens de *spam* que correspondem a 89% de todas as mensagens de correio electrónico observadas.

Capítulo 3 Sistema de Detecção de Intrusões

Com a proliferação dos acessos à internet e com um número crescente de serviços que dispomos enquanto ligados à rede, também o risco de ataques por utilizadores menos bem-intencionados aumentou.

Desta forma é necessário que as ferramentas e mecanismos de segurança acompanhem a constante evolução.

O IDS é uma ferramenta de segurança (*software* ou *appliance*) que tem como função detectar intrusões através da monitorização do tráfego de dados, lançando alarmes e informando caso existam acções que violem a segurança do sistema. São utilizados em conjunto com as típicas Firewall em organizações que pretendam acrescentar mais segurança aos seus sistemas e redes.

Os sistemas IDS são tipicamente constituídos por:

- *Sensores* que geram os eventos e alarmes de segurança;
- *Consola* que controla os sensores, monitoriza os eventos e alertas;
- *Motor* que utiliza as regras de segurança para gerar os alertas a partir dos eventos de segurança;

Nesta secção vão ser analisadas as principais características que distinguem os diferentes tipos de IDS.

3.1 Firewall vs IDS

Uma Firewall tipicamente filtra as entradas e saídas na rede, escolhendo os pacotes que podem ou não passar consoante as suas características, enquanto os sistemas de IDS têm a função de reconhecer potenciais ataques em tempo real, através da análise de todos os processos e padrões de comportamento de uma intrusão. Para tal, o sistema de IDS deve ser capaz de monitorizar e analisar o tráfego da rede, podendo desta forma

reconhecer padrões de ataque ou comportamentos anormais. Uma boa analogia para compreender a diferença entre estes dois sistemas de protecção é pensar na firewall como o gradeamento de uma casa e o sistema de IDS como o alarme da casa[2].

3.2 Tipos de IDS

Existem actualmente diferentes tipos de IDS, estes podem ser classificados por algumas características funcionais, tais como a sua Fonte dos Eventos, Método de detecção, Frequência de utilização, Fiabilidade e a sua Reactividade[6].

3.2.1 Fonte de eventos

Nos sistemas IDS os dados podem usar duas fontes de eventos:

- *A rede (Network based ou NIDS)* onde é monitorizado todo o tráfego.
  São equipamentos na rede que analisam o tráfego e verificam se os pacotes capturados estão dentro de padrões pré-determinados ou não.

Neste tipo de IDS é necessário um posicionamento estratégico do computador na rede para que todo o tráfego que circula na rede possa ser analisado.

- *As máquinas (Host based ou HIDS)* onde é monitorizado o estado dos vários componentes da máquina, hardware, sistema operativo ou aplicações, através de um agente em cada *host*. Este tipo de IDS tem a grande vantagem de estar mais próximo dos recursos alvo de ataques e intrusões, mas a desvantagem de não ter uma visão global das máquinas, mas apenas da máquina que estão a analisar.

Pode ainda haver casos de sistemas IDS que podem actuar quer ao nível da rede, quer ao nível das máquinas, chamados híbridos.

3.2.2 Método de Detecção

A detecção de intrusões pode ter duas variantes:

- *Baseada em Conhecimento* (também designado por método baseado em assinaturas) analisa a actividade em busca de padrões de ataque ou de intrusão conhecidos. É um método bastante eficiente porque a taxa de falsos positivos é baixa. A sua limitação reside no facto de assumir que tudo o que não conhece não é perigoso.

- *Baseada em comportamento* detecta desvios à normalidade no comportamento dos utilizadores ou grupo de utilizadores. Este tipo de detecção tem como principal vantagem a possibilidade de detectar novas formas de ataques e intrusões. Por sua vez, este método tem dois grandes problemas: a definição do que é o comportamento normal do sistema em análise e a geração de muitos falsos positivos se mal configurado.

3.2.3 Frequência de utilização

Podemos ter sistemas IDS que estão constantemente a monitorizar os sistemas, para que desta forma os ataques sejam detectados em tempo real, ou apenas quando existe suspeita de intrusão, por parte de outros mecanismos de segurança sejam eles gráficos *mrtg* (*Multi router traffic grapher*), *logs* de *firewall* ou agentes de monitorização do estado dos serviços, que detectam que algo não se encontra dentro da normalidade.

3.2.4 Fiabilidade

A fiabilidade de um IDS pode ser medida de várias formas. Pode ser medida pelo número de falsos positivos (existe o lançamento de um alerta para uma intrusão que na realidade é inexistente), falsos negativos (quando efectivamente houve uma intrusão mas esta não foi detectada), pela capacidade de conseguir ou não processar todos os eventos ou pela qualidade da informação que consegue obter associada a cada evento.

3.2.5 Reactividade

Após a detecção e identificação de uma intrusão, o sistema IDS pode despoletar alarmes e notificações (IDS Passivo). Os IDS reactivos actuam de forma activa podendo:

- Isolar o recurso atacado.
- Bloquear através da firewall ou router, pacotes provenientes do intruso.
- Tomar acções de contra-ataque que passam por tentar saber quais os serviços que estão a correr na máquina atacante e explorar falhas de segurança.

3.3 Honey Pot

Outro tipo de sistemas IDS são os *honeypot* (ou *honeytrap*) é o tipo de sistema que simula falhas de segurança no sistema, atraindo o "hacker" para esta zona específica do sistema permitindo recolher informação sobre o invasor.

Este tipo de sistema não oferece nenhuma protecção real ao sistema, permite apenas induzir os ataques à zona monitorizada visto que oferece menos segurança que as outras zonas do sistema[7].

3.4 Principais características de um IDS

Para a escolha de um sistema de IDS adequado temos de ter em conta algumas características importantes que o sistema deve ter:

- **Simplicidade de configuração** - Para evitar desperdício de tempo e recursos um IDS deve ser simples e rápido de se configurar.
- **Simplicidade de administração** - Pretende-se que seja uma ferramenta simples de administrar e de interpretar, de forma a que cada alerta lançado possa ser facilmente compreendido.
- **Independência de operação** - A ferramenta tem de ser capaz de se manter activa, desempenhando as suas funções, sem a necessidade de intervenção humana.
- **Tolerância a falhas** - Sendo uma ferramenta que está a operar sem supervisão humana, espera-se que esta seja capaz de, em situação de erro, recuperar e retomar a sua operação normal.
- **Segurança** - O sistema deve ser seguro, para que ataques dirigidos a ele não tenham sucesso. Deve garantir também que não existam serviços a correr na mesma máquina que possam vir a ser explorados para uso num ataque directo ao sistema.
  - **Baixo impacto no funcionamento do sistema/infra-estrutura** - Os sistemas IDS vão funcionar como monitores de rede, capturando e analisando o tráfego. Estas operações não devem prejudicar o normal funcionamento. Assim não devem ser instalados em máquinas de produção visto que as operações executadas geram uma grande carga de processamento.
- **Analisar padrões** - O sistema deve ser capaz de separar e analisar o tráfego e reconhecer padrões de comportamento que possam indiciar um ataque.
- **Não detectável** - Um sistema IDS deve ser discreto na rede de forma a que um atacante não se aperceba que está a ser monitorizado.
- **Resistente a erros de monitorização** - Existem 3 tipos de erros que são susceptíveis de acontecer num sistema de IDS: Falsos Positivos, Falsos Negativos e erros de subversão.

3.5 Limitações de um sistema IDS

Existem algumas limitações nas ferramentas IDS que muitas vezes alteram o comportamento da infra-estrutura onde estão instaladas.

Alguns dos factores a ter em conta são:

- **Estrutura da rede** - Com o crescimento das redes empresariais estas tornaram-se mais complexas e mais segmentadas com diferentes zonas. Desta forma é complicado colocar um NIDS de forma a abranger toda a rede. Para ultrapassar esta limitação é necessário um planeamento cuidado para que o NIDS tenha acesso a todo o tráfego da rede.
- **Limitação de recursos** - O IDS necessita ter recursos suficientes de processamento, memória e armazenamento para análise do tráfego. Caso contrário poderá alterar o normal funcionamento da rede ou poderá ser alvo de ataques DOS.
- **Ataques contra o IDS** - Uma ferramenta IDS mal configurada ou demasiado exposta numa rede faz com que seja um alvo de ataques. Usando técnicas de cifração de dados ou VPN com dados adquiridos de forma ilegal podem servir para ultrapassar um IDS. Existem formas mais complexas de deturpar a verificação do IDS. Se o atacante sabe da existência do sistema de IDS e sabe que tipo de pacotes o IDS está à procura, o atacante pode gerar uma grande quantidade de pacotes suspeitos para que o sistema gere um grande conjunto de falsos positivos. A análise de todos esses alertas irá requerer muito esforço para identificar um possível ataque.

3.6 Ferramentas IDS

Com a impossibilidade de testar todas as ferramentas disponíveis, a escolha feita teve como base as ferramentas de IDS mais comuns disponíveis online e que reuniam um conjunto de características que necessitamos, como o facto de ser uma ferramenta open-source e a necessidade das ferramentas reunirem características diferentes de forma a poderem ser testadas as várias vertentes na detecção de intrusões.

De seguida são analisadas algumas ferramentas de detecção de intrusões.

3.6.1 OSSEC

Ossec é uma ferramenta *open-source* de IDS baseada no computador (*host based*), faz análise de logs, verificação de integridade, monitorização de registos (Windows), detecção de *rootkits*, alertas em tempo real e é pro-activo.

É compatível com a grande maioria dos sistemas operativo tais como Linux, Windows, Mac OS X , FreeBSD e Solaris][9].

3.6.2 Snort

Snort é um dos softwares mais conhecidos na detecção e prevenção de intrusões.

É um sistema de IDS em que o seu alvo é a rede (network based). O Snort é capaz de fazer análise de protocolos, verificação de conteúdos, detecta um conjunto de ataques e sondas, tais como *buffer overflow*, *stealth port scans*, ataques a aplicações web, sondas SMB e OS *fingerprinting*[10].

3.6.3 AntiSniff

O AntiSniff é um IDS que permite a detecção de máquinas que se encontram a analisar o tráfego da rede. Esta ferramenta provoca estímulos numa máquina inspectora de modo a que esta revele a sua actividade camuflada [11].

3.6.4 TripWire

Tripwire é um IDS baseado no computador, funciona essencialmente para detectar alteração de ficheiros em sistemas UNIX, permite que os ficheiros críticos do sistema sejam constantemente monitorizados, e caso algum ficheiro seja apagado ou alterado o administrador é informado de forma a poder corrigir o problema.

Para isto acontecer o Tripwire guarda checksums , o tamanho exacto dos ficheiros e outros dados sensíveis relativos aos ficheiros[12].

3.6.5 Valhala

Vallhala é um sistema de IDS baseado num *honeypot*, permite criar simular servidores de *Web, Ftp, finger, smtp, pop3, echo, dytime, tftp* e *portforwarding*.

"Simula portas de trojans conhecidos (como Netbus, subseven, etc) e ainda possibilita utilizar outras portas"[13].

3.6.6 Tabela comparativa

Após a escolha de algumas das ferramentas disponíveis foi necessário recolher as características de cada uma delas de forma a verificar a compatibilidade com o ambiente em que o sistema se vai inserir. De seguida é apresentada a tabela comparativa entre os vários IDSs analisados [Tabela 3.1].

Nome	Tipo	S.O. Servidor	S.O . Agente	Tipo de Gestão
Snort	NIDS	Linux		Independente
OSSEC	HIDS	Linux	Linux/Windos/BSD	Centralizada
AntiSniff	NIDS	Linux		Independente
Tripwire OpenSource	HIDS	Linux	Linux/Windows	Centralizada
Valhala	Honeypot	Windows		Independente

Tabela 3.1: Tabela Comparativa entre os vários IDS

Capítulo 4 Rede FCUL

Neste capítulo irá ser feita uma análise à infra-estrutura, serviços e segurança dos sistemas geridos pelo Centro de Informática.

4.1 Estrutura da rede

A Faculdade de Ciências faz parte da Universidade de Lisboa, dividida por vários departamentos e unidades organizacionais, mais ou menos independentes e autónomas. Na figura 7 encontra-se uma representação lógica da estrutura da rede da Faculdade Ciências.

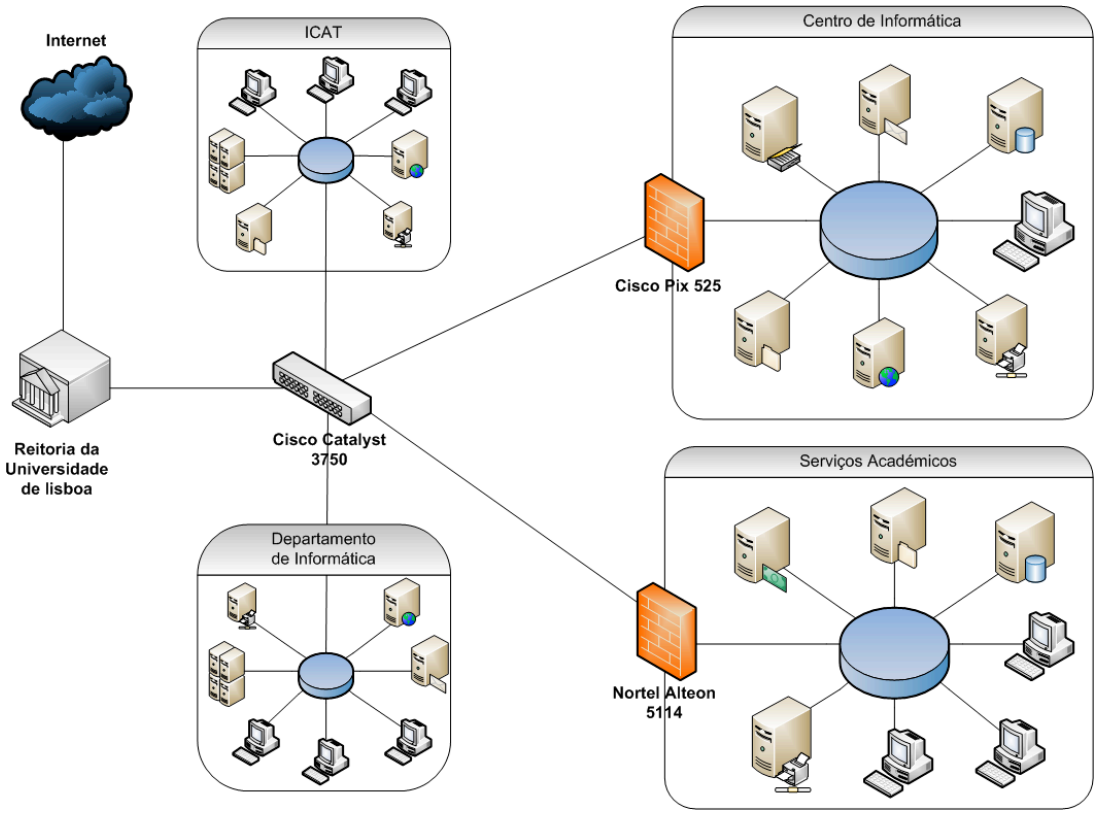

Figura 4.1: Rede FCUL

4.2 Estrutura Física

Todos os servidores e grande parte dos equipamentos de rede geridos pelo centro de informática encontram-se num *Data Center* localizado no edifício C1 da Faculdade de Ciências da Universidade de Lisboa.

Para melhor gestão, manutenção e redução dos custos energéticos grande parte dos servidores são máquinas virtuais *VMware*®.

Ainda são utilizadas algumas máquinas físicas como medida de prevenção no caso de falha da estrutura de suporte às máquinas virtuais.

4.3 Serviços de Rede

A infra-estrutura do Centro de Informática fornece um conjunto de serviços essenciais para o funcionamento da instituição.

Estes serviços são divididos por dois diferentes domínios na Faculdade de Ciências, o domínio *fc.ul.pt* destinado aos servidores e aos computadores dos funcionários docentes e não-docentes da instituição e o domínio *alunos.fc.ul.pt* destinado aos computadores das salas de aulas e ao Espaço Estudante.

4.3.1 Domain Controller

O Domain Controller é o servidor que permite a criação e gestão de domínios.

Juntamente com o Domain Controller existe a A*ctive Directory* que é uma estrutura essencial do d*omínio,* necessária para o funcionamento de um grande conjunto de serviços na Faculdade de Ciências. Nela estão inseridos todos os docentes, investigadores, funcionários e alunos da instituição. Serve para autenticação de serviços como o correio electrónico, Single Sign-On, *eduroam (rede* sem fios) e para o acesso a qualquer computador que esteja no domínio *fc.ul.pt* ou *alunos.fc.ul.pt*.

O domínio *fc.ul.pt* tem 3 Domain Controllers sendo que os dois primeiros são máquinas virtuais e o terceiro uma máquina física, que têm com sistema operativo o Microsoft Windows Server 20003.

Os servidores que operam como Domain Controller para o domínio *fc.ul.pt*, são os seguintes:

- *fc-winsrv01.fc.ul.pt*
- *fc-winsrv02.fc.ul.pt*
- *fc-winsrv03.fc.ul.pt*

4.3.2 DHCP

Um dos serviços elementares disponibilizados é o DHCP, que permite que todos os computadores e equipamentos que estão ligados à rede tenham um endereço de IP e possam comunicar.

Existem dois servidores de DHCP, ambos com o sistema operativo Microsoft Windows Server 2003, um para a rede dos alunos e outro para a restante rede.

Sendo que o servidor DHCP para o domínio alunos é o *alunos-winsrv02.alunos.fc.ul.pt* e o servidor DHCP para o domínio FC é o *fc-winsrv02.fc.ul.pt*

4.3.3 DNS Interno

Este serviço permite fazer a tradução de nomes para IPs e é essencial para o acesso a máquinas apenas pelo nome e a páginas de internet que apenas estão disponíveis dentro da rede FCUL.

Existem 6 servidores Windows Server 2003 com a função de DNS interno, 3 para o domínio *fc.ul.pt* e 3 para o domínio *alunos.fc.ul.pt.*

Sendo que para um dos domínios existem 2 servidores virtuais e 1 físico.

Para o domínio *fc.ul.pt* temos os seguintes servidores:

- *fc-winsrv01.fc.ul.pt*
- *fc-winsrv02.fc.ul.pt*
- *fc-winsrv03.fc.ul.pt*

Para o domínio *alunos.fc.ul.pt* temos os seguintes servidores:

- *alunos-winsrv01.alunos.fc.ul.pt*
- *alunos-winsrv02.alunos.fc.ul.pt*
- *alunos-winsrv03.alunos.fc.ul.pt*

4.3.4 DNS Externo

Para o acesso a páginas de internet externas necessitamos também de um servidor de DNS que nos faça a tradução de nomes para IPs. Para tal a FCUL dispõem de dois servidores instalados com Red Hat Enterprise 5.

Os servidores de DNS externos são os seguintes:

- *ns01.fc.ul.pt*
- *ns02.fc.ul.pt*

4.3.5 Servidor de correio electrónico

O servidor de correio electrónico é constituído por um *cluster* de duas máquinas aliadas a uma *storage* para o armazenamento do correio electrónico.

O serviço é fornecido por uma máquina com *Windows Server 2003 R2 Standard x64 edition* e o *Microsoft Exchage Server 2007*.

4.4 Serviços Web

A Faculdade de Ciências possui muitos serviços que são disponibilizados via internet.

4.4.1 Gestão FCUL

Este serviço centraliza um conjunto de ferramentas de gestão para alunos e pessoal docente e não docente e está acessível através da página *gestao.fc.ul.pt*.

Através deste serviço é possível aos alunos fazer a gestão da sua conta pessoal, verificar o pagamento das propinas e gerir as suas impressões.

Para um docente é possível para além da gestão da sua conta pessoal pode pedir a activação de tomadas ou a criação de páginas de internet.

4.4.2 Moodle FCUL

O Moodle é a plataforma de E-learning da Faculdade de Ciências, nesta plataforma estão inseridos conteúdos de mais de 300 disciplinas incluindo alguns projectos internacionais, sendo um serviço importante para o normal funcionamento da instituição.

4.4.1 Webmail FCUL

Este serviço web permite o acesso à conta de correio electrónico da Faculdade de Ciências e está acessível através da página *webmail.fc.ul.pt.*

Através deste site é possível gerir a conta de correio electrónico sendo possível a criação de pastas e regras para as mensagens.

4.4.2 Single Sign On FCUL

A figura 4.2 mostra a implementação de mais um dos serviços importantes para a FCUL, o sistema de Single Sign On na FCUL, serviço este que serve de autenticação única para as Páginas da FCUL. Neste momento encontra-se a funcionar para a página moodle*.fc.ul.pt, ci.fc.ul.pt, gestão.fc.ul.pt e gestão.alunos.fc.ul.pt*. A autenticação pode ser feita através da conta FC, AlunosFC ou através do cartão do cidadão.

Usa dois servidores *Red Hat Enterprise Linux Server Release 5.4*, de forma a poder fazer o balanceamento de carga dos pedidos de autenticação, uma base de dados MySql para guardar os *tickets* de sessão e usa a Active Directory da FCUL para verificar a autenticação.

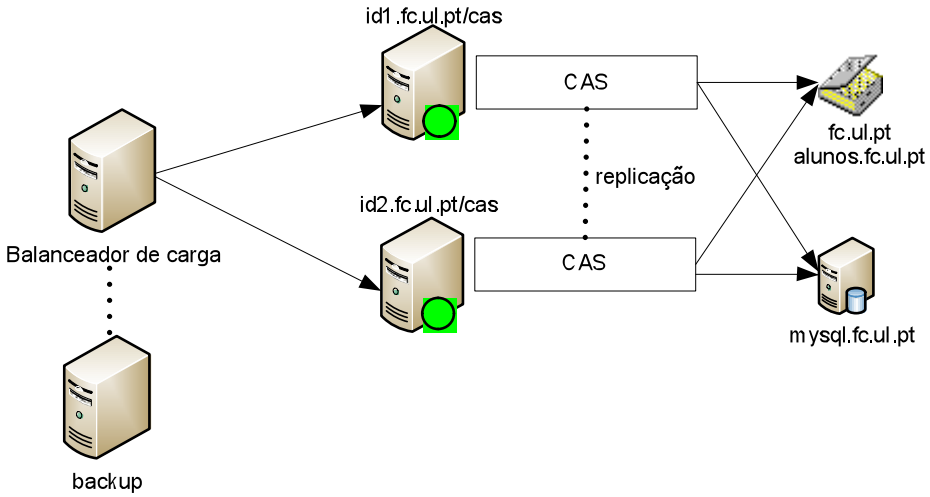

Figura 4.2: Diagrama do SSO da FCUL

4.5 Segurança

A segurança de toda a infra-estrutura gerida pelo Centro de Informática é feita através de um conjunto de mecanismos de segurança.

Este mecanismo passa pelos equipamentos e sistemas de segurança, políticas e procedimentos e a restrição ao acesso físico ao meio.

4.5.1 Sistemas de Segurança

A estrutura de segurança existente é constituída por duas *firewalls*, uma Cisco PIX 525[14] localizada à entrada da rede FCUL e outra Nortel Alteon 5114[15] à entrada da rede dos Serviços Académicos.

A estrutura da rede e servidores está configurada para que:

- Exista uma separação da rede em várias redes virtuais (*vlans*), com privilégios diferentes.
- O acesso à rede FCUL através da rede sem fios ou de VPN necessita de autenticação, que apenas é atribuída a alunos, docentes, investigadores e funcionários.
- Para os servidores públicos que precisam obrigatoriamente de aceder à rede interna, estes acessos são feitos por meio de uma proxy.

4.5.2 Políticas e procedimentos

O Centro de Informática possui políticas e procedimentos que devem ser seguidos por todos aqueles que têm um papel importante na manutenção da infra-estrutura informática, que são os seguintes:

- Cada novo equipamento que se pretenda ligar à rede necessita de ter o seu *Mac-Address* registado.
- Todas as tomadas de rede necessitam de registo para serem utilizadas.
- Nenhum equipamento de rede pode estar registado em duas tomadas.
- Todos os servidores que tenham um serviço disponibilizado para o exterior não podem ter outro dispositivo de rede para acesso à rede interna.
- Não é permitida a criação de pontos de acesso wireless com ligação à rede da FCUL.

4.5.3 Segurança Física

Para garantir mais segurança é necessário controlar o acesso físico e lógico às infra-estruturas de suporte aos serviços disponibilizados.

O acesso físico ao *Data Center* está limitado aos trabalhadores do Centro de Informática sejam eles Operadores ou Técnicos. Qualquer operação de limpeza ou manutenção feita por elementos externos ao serviço são feitas com supervisão. Existe um projecto de implementação de acesso por *RFID* ao *Data Center* de forma a ser possível uma melhor monitorização e controlo dos acessos.

Para o acesso aos bastidores espalhados ao longo dos campos é necessária uma chave que apenas está disponível nas instalações do Centro de Informática ou através da Central de Segurança.

Todo o Campus da Faculdade de Ciências está coberto por um sistema de Vigilância, que faz com que qualquer tentativa de acesso, não autorizado às infra-estruturas seja gravada.

Capítulo 5

Sistema de Detecção de Intrusões FCUL

Após a análise das infra-estruturas geridas e das soluções disponíveis, a opção escolhida foi uma combinação de duas ferramentas OSSEC e SNORT. Neste capítulo é explicada a razão desta escolha e qual o funcionamento da mesma.

5.1 Requisitos da solução IDS a implementar

Para a escolha da melhor solução IDS a implementar na FCUL foi preciso verificar a compatibilidade das soluções com a infra-estrutura que queremos monitorizar.

De acordo com essa análise as características que a solução deve apresentar são as seguintes:

- **Hybrid IDS** - Devido à complexidade da rede e à grande quantidade de tráfego que por ela circula a solução recai num sistema IDS Híbrido, ou seja, a solução possui uma componente *Host Based* (OSSEC) e outra componente *Network Based* (SNORT). Com esta solução híbrida temos uma solução eficiente na protecção de uma máquina específica e vamos ao mesmo tempo analisar o tráfego que circula nas máquinas, eliminado algumas das limitações que cada uma das soluções teria se fossem implementadas individualmente, tornando o sistema mais robusto.
- **Suporte Windows e Linux** - A infra-estrutura gerida pelo Centro de Informática inclui servidores Windows Server e Linux RedHat, logo é necessário que seja possível monitorizar ambos os sistemas operativos sem qualquer limitação.

- **Escalável** - Como a infra-estrutura do Centro é dinâmica e sempre em crescimento temos de ter em conta a escalabilidade da solução para que esta possa ser usada mesmo com a inclusão de novos equipamentos.
- **Gestão Centralizada -** O número de máquinas a monitorizar e controlar é demasiado grande para que a gestão máquina a máquina seja viável, logo a solução passa por uma gestão centralizada de todas as máquinas.

5.2 Arquitectura OSSEC

A arquitectura do OSSEC é composta por um servidor central que monitoriza tudo, recebe informação dos vários agentes, informação *syslog*, bases de dados e informação de dispositivos que não tenham um agente instalado, características que se enquadram nos requisitos de um sistema de IDS para a realidade da infra-estrutura gerida pelo Centro de Informática.

Os elementos que compõem a solução são:

- **Servidor Central** - É responsável pelo armazenamento dos ficheiros de verificação da integridade das bases de dados, *logs*, eventos e entradas dos sistemas de auditoria. Todas as regras, *decoders* e as opções de configurações gerais são armazenadas centralmente no servidor, permitindo uma administração mais fácil mesmo para um grande número de agentes.

- **Agentes** - Um agente é um pequeno programa instalado no sistema que queremos monitorizar. Este recolhe informação em tempo real e envia-a para o Servidor Central para análise. Necessita de pouca memória e processamento o que faz com que não influencie o normal funcionamento do sistema.

- ***Firewalls*, *switches* e *routers***- O OSSEC permite receber e analisar informação proveniente dos eventos syslog de uma grande variedade de *firewalls, switchs* e *routers*. Suporta todos os routers Cisco, *firewalls* Cisco

(PIX), Cisco FWSM, Cisco ASA, routers Juniper, firewalls Netscreen, Checkpoint e muitos mais.

O OSSEC consegue interpretar logs de múltiplas fontes por exemplo: *sshd(OpenSSH), Samba, Proftpd, Microsoft FTP server, Imapd, Postfix, Sendmail, vpopmail, Microsoft Exchange, Apache, IIS5, IIS6, Iptablesm NetScreen, Snort, Nmap, Symantec AV, Arpwatch, Named, Squid, Windows event logs*, entre muitos outros, fazendo com que seja uma ferramenta compatível com qualquer infra-estrutura.

5.2.1 Processos Internos

Internamente o sistema é composto por um conjunto de processos (*deamons*), cada um com uma tarefa específica:

- **Analysid** - Processo principal que faz toda a análise.
- **Remoted** - Recebe os *logs* remotos dos agentes.
- **Logcollector** - Lê os ficheiros de logs (*syslog, Windows event logs, IIS,* etc).
- **Agentd** - Processo no agente que envia os *logs* para o servidor.
- **Maild** - Responsável pelo envio de e-mails de alerta.
- **Execd** - Executa as respostas activas.
- **Monitord** - Responsável pela monitorização do estado dos agentes, compressão e assinatura dos ficheiros de *log*.
- **Ossec-control** - Permite iniciar e parar todos os processos anteriores.

5.2.2 Fluxo dos ficheiros de log

Desde que são produzidos até ao momento em que podem gerar um alerta um ficheiro de log passa por um conjunto de processos como demonstra a figura 10.

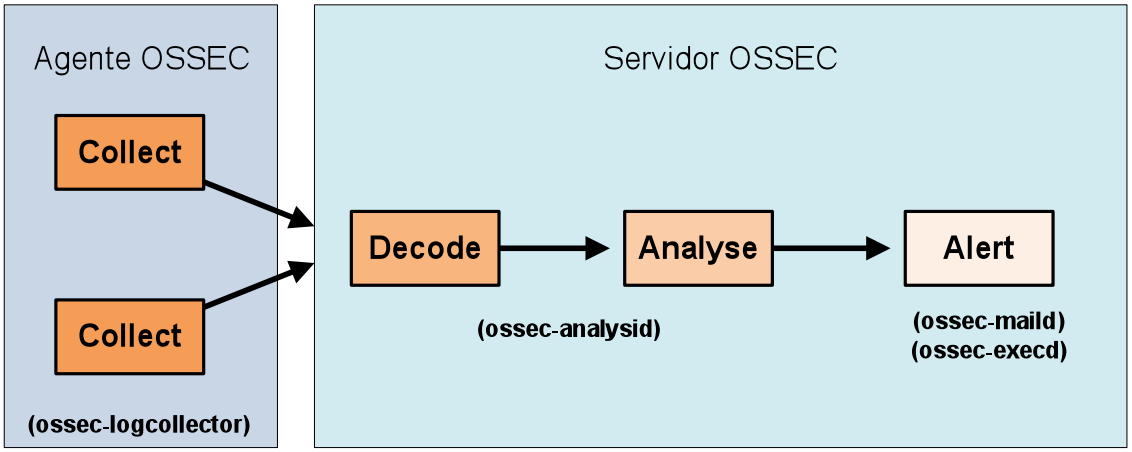

Figura 5.1: Fluxo dos ficheiros de log

- Nos agentes a recolha dos *logs* é feita pelo *ossec-logcollector*.

- Após a chegada ao servidor a análise e descodificação são feitas pelo *ossec-analysisid.*

- Os alertas são lançados pelo *ossec-maild*.

- As respostas activas são feitas pelo *ossec-execd*.

5.2.3 Análise dos logs

O processo responsável pela análise dos *logs* recebidos é o *analysisd*, é divido em 3 partes o *pre-decode, decode* e *rule matching*, como mostra a figura 11.

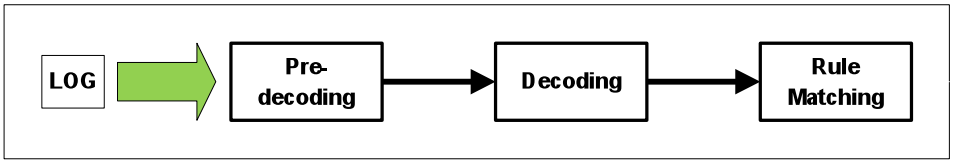

Figura 5.2: Fluxo interno dos ficheiros de log

Cada uma destas partes é responsável por uma tarefa distinta no tratamento dos logs.

- Pre-decode - É o processo responsável por extrair a informação simples enviada através de protocolos como o *Syslog* e o *Apple System Log*, preenchendo os seguintes campos (Tabela 2):

Campo	Descrição
Hostname	Hostname da origem do evento
program_name	Nome do programa que deu origem ao evento
log	Descrição do evento
time/date	Hora e Data do evento

Tabela 5.1: Parâmetros da função de Pre-decoding

Exemplo:

Log original:

```
2010 Aug 13 05:01:29 ubuntu syslogd: restart
```

Após pre-decode:

```
time/date -> 2010 Aug 13 05:01:29
hostname -> ubuntu
program_name -> syslogd
log -> restart
```

- **Decoding-** É responsável por identificar a informação proveniente dos logs podendo preencher os seguintes campos (Tabela 3):

Campo	Descrição
log	Hostname da origem do evento
full_log	Evento completo
location	Origem do log
hostname	Hostname da origem do evento
program_name	Hostname da origem do evento
scrip	Endereço de IP da origem do evento
dstip	Endereço de IP de destino do evento
srcport	Porto de origem do evento
dstport	Porto de destino do evento
protocol	Protocolo do evento
action	Acção tomada pelo evento
srcuser	Utilizador que deu origem ao evento
dstuser	Utilizador de destino do evento
id	ID referente ao evento
status	Estado do Decoded
command	Comando executado pelo evento
url	O URL do evento
data	Dados adicionais que possam ser extraídos do evento
systemname	Nome do sistema que originou o evento

Tabela 5.2: Parâmetros da função de Decoding

Exemplo de um decoding:

A informação original do log :

```
   2010 Aug 14 16:27:41 ubuntu sshd[1025]: Accepted
password for root from 10.101.5.85 port 22 ssh
```

A informação após decoding:

```
   time/date -> 2010 Aug 13 05:01:29
   hostname -> ubuntu
   program_name -> syslogd
   log -> Accepted password for root from 10.101.5.85
port 22 ssh
   srcip -> 10.101.5.85
```

- **Rule Matching** - Após a execução dos processos de *Decoding* o processo de *rule matching* vai verificar se existe alguma regra associada à informação que foi recolhida pelos logs.

Existem dois tipos de regras:

- Atómicas - São baseadas num só evento.
- Compostas - São baseadas no conjunto de eventos encadeados.

As regras são criadas em XML, o que torna simples a interpretação e a criação de novas regras.

Cada regra deve conter a tag "*rule id"* (um inteiro que identifica a regra), "*level*" que indica o nível de alerta (sendo 0 o mais baixo e 15 o mais alto) e o padrão que pode ser um qualquer elemento retirado do log como o *user, hostname*, padrão textual do log etc.

Exemplo da representação XML de uma regra:

```
    <rule id="110" level="5" >
    <decoded_as>sshd</decoded_as>
    <description>Logging    every    decoded    sshd    message
</description>
    </rule>
    <rule id="120" level="7">
    <if_sid>110</if_sid>
    <match>^Failed password</match>
    <description>Failed password attempt</description>
    </rule>
```

5.2.4 Segurança interna

Esta divisão de tarefas em processos é feita a pensar na segurança do próprio sistema.

Desta forma cada um dos processos é executado por um *user* diferente com privilégios próprios, todos executados com a operação *chroot* para que desta forma não seja possível aceder fora desse directório.

-**Analysid** - Executado como user ossec.

-**Remoted** - - Executado como user ossecr

-**Maild** - - Executado como user ossecm

-**Logcollector** - Executado como root, mas apenas lê os ficheiros de log mas não os analisa.

-**Agentd** - Executado nos agentes como user ossec.

5.2.5 Comunicação

Estando os agentes dispersos pelas várias máquinas na rede a comunicação dos agentes e o servidor é feita por UDP no porto 1514 ou através do *syslog* porto 514 (figura 12).

A comunicação por sua vez é cifrada (*BlowFish)* usando chaves partilhadas, que são geradas pelo servidor e inseridas no agente quando este é instalado.

A informação recebida através do *syslog* não é cifrada, o que, por motivos de segurança pode ser desactivada.

Os dados transmitidos são comprimidos usando zlib.

A comunicação feita é independente da plataforma em que o agente se insere.

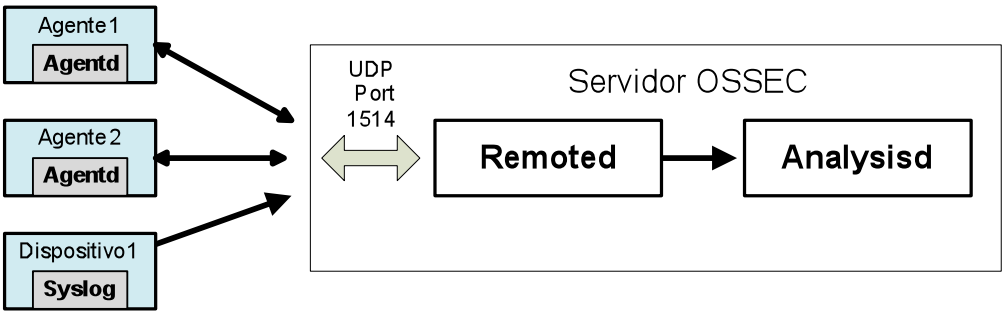

Figura 5.3: Fluxo dos ficheiros de log

5.2.6 Active Response

Esta função permite que haja uma acção em resposta a uma ameaça ou violação das políticas estabelecidas, despoletando um *script* que executa uma determinada tarefa.

Por omissão esta função contém alguns *scripts* de resposta activa (*active-response/bin*).

- disable-account.sh - Desactiva uma conta de utilizador especifica
- host-deny.sh - Adiciona um determinado host ao ficheiro */etc/hosts.deny* que é usado pelo *tcpwrappers*.
- route-null.sh - Adiciona um determinado IP à lista de IP's rejeitados ou à lista de *blackhole*. Este *script* é usado principalmente para máquinas que estejam a ser usadas como encaminhadores de pacotes.
- firewall-drop.sh - Este script adiciona um endereço IP à drop list da iptable, bloqueando o seu acesso.
- ipfw_mac.sh - Scrip firewall-drop para Mac OS X
- ipfw.sh - Script firewall-drop para *ipfw* (filtro de pacotes IP do FreeBSD)
- pf.sh - Script firewall-drop para *pf* (filtro de pacotes do BSD)
- ossec-tweeter.sh - Permite enviar um tweet com a informação de um log

Cada um destes scripts é executado como *root* na máquina de destino desta forma podendo executar qualquer comando ou conjuntos de comando que um administrador do sistema executaria.

Para podermos criar novas *active responses* é necessário adicionar uma nova entrada (XML) <*command*> e outra <*active_response*> no ficheiro *ossec.conf* (*ossec/etc)*.

A entrada <*command*> deve conter as seguintes tags.

- <name> - Nome do comando (normalmente associada ao script ou executável que vai ser despoletado).
- <executable> - Nome do script ou executável (tem de existir *active-response/bin*)
- <expect> - Lista de variáveis separadas por vírgulas que vão ser usadas pelo script (srcip e/ou user).
- <timeout-allowed> - Pode conter *yes* ou *no* , caso seja *yes* terá um tempo limite para executar, caso o tempo expire o executável deve ser capaz de repor as alterações feitas.

A entrada <*active_response*> deve conter as seguintes tags.

- <disable> - Pode conter "*yes"* ou "*no"*, caso pretendamos activar ou desactivar uma active response.
- <command> - Deve conter o <name> definido num <command>
- <location> - Pode conter *local*(para server executado localmente), *server*(para ser executado no servidor), *defined-agent*(para ser executado num agente expecifico) ou *all*(para ser executado em todos os agentes) .
- <agent_id> - Caso a opção anterior tenha sido *agent* , temos de indicar o "*id"* do agente.
- <level> - Indica em que grau mínimo de ameaça deve ser accionada esta *active response* .
- <rules_id> - Deve ser indicada o "*id*" de uma *rule* (ou múltiplas *rules*), que queremos que seja accionada esta *active response* .
- <rules_group> - Indica o grupo de regra que acciona esta *active response* .
- <timeout> - Duração em segundos do tempo que o script ou executável poderá executar (por exemplo bloquear um *host* por 120 segundos).

Como podemos reparar as *active responses* estão mais direccionadas para sistemas UNIX/Linux visto que através da linha de comandos podemos controlar todo sistema.

As *active responses* vêm por omissão desactivadas nos agentes Windows por isso deverão ser activadas alterando o *ossec.conf* no agente.

Os scripts executados em máquinas Windows deverão ser *.cmd* ou *.bat* .

5.2.7 Verificação de Integridade

Esta função pode ser integrada em qualquer tipo de instalação do OSSEC, seja ela servidor, agente ou instalação *stand-alone*.

Com esta função activa é possível verificar a integridade dos ficheiros de sistema de forma a detectar que estes não são alterados ou danificados.

É capaz de monitorizar os ficheiros de configuração e executáveis em Unix, Linux ou BSD (/etc, /bin,...) ou em Windows ("\Windows\System32" e algumas chaves de registos).

A configuração desta opção é dividida em 3 partes:

- <*frequency*> - Onde é possível definir com que frequência é que as directorias são verificadas.
- <*directories*> - Onde são especificadas as directorias a monitorizar e onde podemos definir se pretendermos a verificação apenas por algum parâmetro especifico, por exemplo a alteração de tamanho ou alteração de privilégios.
- <*ignore*> - Permite especificar ficheiros que não queremos incluir na verificação.

5.3 Arquitectura SNORT

O Snort é um sistema de detecção de intrusões baseado na rede. É um IDS baseado em assinaturas que usa regras para verificar a existência de pacotes com informações que possam indiciar a ocorrência de um ataque. Regras essas que consistem num conjunto de requisitos que gerariam um alerta.

5.3.1 Componentes SNORT

Na figura 5.4 podemos ver como se processa a análise dos pacotes recolhidos pelo Snort e os quatro principais componentes que intervêm nesse processo.

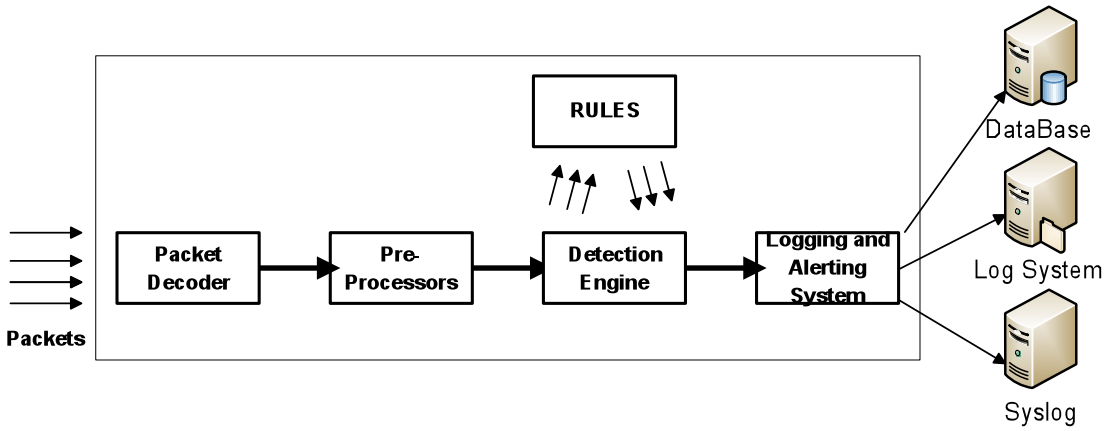

Figura 5.4: Fluxo de processamento do Snort

Os quatro componentes principais do Snort são:

- *Packet Decoder* – É responsável pela captura dos pacotes das diferentes interfaces de rede e a preparação para o pré-processamento.
- *Pre-processors* – São responsáveis pela decomposição e classificação dos pacotes para posteriormente serem usados pelo mecanismo de detecção (*Detection Engine)*. Pode analisar, descartar, modificar e/ou gerar um alerta dependendo do tipo de pacote. Neste pré-processamento é possível detectar ataques como *port scanning* ou *ARP spoofing*.

- *Detection Engine* – É responsável pela comparação dos dados dos pacotes recebidos pelo pré-processador (*Preprocessors)* com as informações das regras de ataques conhecidos. Caso os dados dos pacotes correspondam à informação de alguma regra a informação é enviada para o sistema de alerta e registo (*Logging and Alerting System*).
- *Logging and Alerting System* – Caso os dados que passam pelo mecanismo de detecção correspondam a alguma regra, são passados ao sistema de alerta e registo (*Logging and Alerting System*). Algumas das acções deste componente são a geração de *log reports*, o envio de *snmp traps*, escrita numa base de dados ou o envio de uma mensagem para um servidor *Syslog*.

Capítulo 6 Implementação e Testes

Neste capítulo explica-se a instalação e configuração do sistema e dos agentes.

Houve uma preparação do equipamento onde o sistema iria ser instalado, a instalação da solução e a respectiva configuração do OSSEC, do SNORT e dos agentes.

Foram feitos testes funcionais, simulando ataques e verificando de que forma é que o sistema IDS reage aos ataques.

Nos testes de impacto verificou-se de que forma é que os agentes influenciam as máquinas onde se encontram instalados.

6.1 Implementação do Servidor IDS

Para a implementação do servidor de IDS foi necessário instalar o OSSEC e o SNORT num servidor e configurá-los de forma a funcionarem em conjunto e a receberem informação dos vários sensores.

Foi também implementado um *front-end,* acessível via Web Browser para a monitorização de todos os agentes espalhados pelas várias máquinas da rede.

6.1.1 Instalação

Para a instalação do OSSEC e do SNORT foi necessária a criação de uma máquina virtual e a instalação de um sistema operativo Linux, o sistema operativo escolhido foi o *Red Hat Enterprise Linux Server release 5.5* (Tikanga).

A escolha do *Red Hat Enterprise* deveu-se a este ser compatível com as aplicações que vão instaladas e de ser o sistema operativo utilizado nos servidores Linux geridos pelo Centro de Informática, já tendo provas dadas de solidez e robustez.

Neste servidor foi necessário instalar um compilador de C (*gcc*) e as *devolpment headers*, para a implementação do *front-end* de gestão foi instalado o Apache Web Server juntamente com o PHP 4.1.

A localização escolhida par a instalação do OSSEC foi */var/ossec,* visto que é uma localização apenas acessível a utilizadores com privilégios de administração.

6.1.2 Configuração OSSEC

Após a instalação no servidor foi necessário configurar alguns parâmetros para que o sistema funcione de forma correcta.

As configurações feitas na aplicação foram as seguintes:

- Configurar a porta 1514 UDP para servir de comunicação entre o servidor e os agentes.
- Configurar o servidor para enviar as notificações por correio electrónico.
- Activadas as *active responses*.
- Adicionado o IP local à *white-list* de forma a indicar que é um *host* confiável.

Para a correcta comunicação entre o servidor e os agentes tivemos de configurar a firewall da máquina de forma a permitir as comunicações na porta UDP 1514, porta esta definida para a comunicação entre o servidor e os agentes. Foi necessário adicionar uma nova regra à Firewall *(iptables*) adicionando a seguinte linha ao ficheiro de configuração (*/etc/sysconfig/iptables*).

```
   -A RH-Firewall-1-INPUT -m state --state NEW -m udp -p udp --dport 1514
-j ACCEPT
```

A explicação dos parâmetros utilizados:

-A RH-Firewall-1-INPUT - Adiciona a regra no fim da cadeia

-- state NEW - Indica que é uma nova ligação.

-**p udp** - Definimos o protocolo UDP a ser usado para a comunicação.

--dport 1514 - Definimos o porta para a comunicação.

-j ACCEPT - Aceita a ligação, caso todos os parâmetros anteriores sejam correspondidos.

6.1.3 Configuração Snort

Para a integração do Snort com o OSSEC foi necessário fazer algumas alterações à configuração inicial do Snort.

Estas alterações permitiram que o OSSEC interpretasse os logs escritos pelo Snort.

Foi necessário alterar o ficheiro de configuração do Snort, de forma a alterar a forma como os ficheiros de log são escritos, passando-se para um formato CSV (*Comma Separated Values*), de forma a poder ser interpretado pelo OSSEC.

```
output alert_csv:/var/log/snort-spooker.log msg,timestamp,src,dst,dstport
```

Depois foi adicionada uma nova entrada no ficheiro de configuração do OSSEC

```
<localfile>
      <log_format>syslog</log_format>
      <location>/var/log/snort/alert</location>
</localfile>
<localfile>
      <log_format>syslog</log_format>
```

Foi necessário acrescentar um novo *decoder* para a nova informação, e para isso alteramos o ficheiro *decoder.xml* acrescentado as seguintes linhas:

```
<decoder name="ossec-snort">
      <prematch>^[OSSEC]</prematch>
      <regex offset="after_prematch">,\S+,(\d+.\d+.\d+.\d+)
,(\d+.\d+.\d+.\d+),(\d+)</regex>
```

6.2 Configuração dos Agentes

Qualquer agente instalado deverá conter uma chave de forma a poder ser identificado pelo servidor.

Estas chaves são geradas pelo servidor e servem para identificar o agente e para cifrar a informação que é partilhada entre os agentes e o servidor.

Desta forma é garantido que não são processadas mensagens ilícitas enviadas por *hosts* desconhecidos.

Após a instalação dos agentes foi necessário utilizar uma ferramenta chamada de *manage_agents* que permite gerar as chaves necessárias para identificar cada agente e registá-los no servidor.

Esta ferramenta permite ainda remover agentes e verificar quais deles é que se encontram em funcionamento.

6.3 Configuração Front-End

Para o *front-end* foi usada uma ferramenta OSSEC Web UI, instalada no servidor OSSEC e acessível via Web Browser.

Foi necessária a configuração do servidor Apache de forma a ser possível ter acesso ao WUI a partir do endereço *ids.fc.ul.pt.*

Para tal, após a instalação do Apache, foi necessário verificar qual o nome do utilizador que estava a utiliza-lo, adicionar esse utilizador ao grupo “ossec” e dar-lhe permissões para aceder à pasta onde o WUI se encontrava instalado.

A partir deste *front end* é possível:

- Ver os agentes disponíveis.
- Ver os últimos ficheiros modificados.
- Ver os últimos eventos registados.
- O acesso ao verificador de integridade de cada agente.
- Consultar tabelas de estatísticas com filtro temporal.
- A procura detalhada de alertas, com filtros por tempo, sensor, grau de alerta, formato do log, padrão, user, rule id, etc.

6.4 Testes

Após a implementação e configuração do servidor e dos agentes, foi necessário testar se as "alarmísticas" estavam a funcionar e qual o comportamento da ferramenta num conjunto de diferentes situações.

Para verificar qual o impacto que os agentes iriam ter nas máquinas hospedeiras foram feitos testes de impacto.

6.4.1 Ferramentas de Teste

Para os testes feitos utilizamos um conjunto de ferramentas que são:

- **BackTrack 4 -** É uma distribuição de linux baseada em Debian, que contém uma grande quantidade de ferramentas úteis para *penetration testing*.
- **NetCat** - É uma ferramenta de análise de rede, que permite fazer *port scanning*, transferência de ficheiros, port listening e pode ser usado para iniciar ligações TCP e UDP.
- **Nmap** -O Nmap é uma ferramenta que permite fazer análise à rede e aos hosts. Permite aí uma pesquisa de forma encontrar os equipamentos ligados à rede. Permite analisar os serviços que estão a correr nas máquinas e que portas estão a ser usadas, desta forma podendo encontrar falhas de segurança.

- **Nikto2 -** É uma ferramenta que permite testes a paginas de internet verificando se existem vulnerabilidades.

  • **Xhydra** - É uma ferramenta que permite fazer ataques de força-bruta contra vários tipos de protocolos como FTP, telnet ou SSH.

6.4.2 Testes Funcionais

Nestes testes são feitos um conjunto de ataques e de modificações nas máquinas. Os dados foram recolhidos através da Interface Web e das notificações recebidas por correio electrónico.

Visto que os servidores onde vão ser instalados já se encontram em produção e para não haver risco de comprometer o seu normal funcionamento foram utilizadas máquinas virtuais, onde se tentou que reflectissem os mesmos serviços disponibilizados nos servidores reais.

Para a realização dos testes foram utilizados computadores com os seguintes sistemas operativos:

- Microsoft Windows Server 2003, com uma Active Directory e Exchange 2003
- Red Hat Enterprise Linux Server, com servidor de Moodle
- Ubuntu 9.04, com FTP Server, Web Server e SSH server.
- Microsoft Windows XP SP2

a) Port Scan

Para este ataque utilizou-se a ferramenta Nmap que nos permite a pesquisa de máquinas na nossa rede e as portas que estão abertas em cada uma delas.

Enquanto se verificou que portas estariam abertas na nossa máquina Ubuntu 9.04 obtivemos o seguinte alerta:

```
   Received From: (ubuntu) 10.101.5.83->/var/log/auth.log
Rule: 5701 fired (level 8) -> "Possible attack on the ssh server (or
version gathering)."
Portion of the log(s):
Sep 28 19:45:03 n00b-desktop sshd[17498]: Bad protocol version
identification 'GET' from 10.101.5.82
```

b) Ataque ao servidor Web

Para este ataque utilizou-se a ferramenta *Nikto2* para pesquisar possíveis falhas no servidor web.

A máquina atacante tem como IP 10.101.5.104 e a vitima a máquina com sistema operativo Ubuntu 9.04 e com IP 10.101.5.83.

Como resultado deste ataque receberam-se os seguintes alertas:

```
    "WEB-ATTACKS mail command attempt",01/16-19:55:16.318213
,10.101.5.104,10.101.5.83,80
    "WEB-ATTACKS /bin/ls command attempt",01/16-19:55:18.511137
,10.101.5.104,10.101.5.83,80
    "WEB-ATTACKS /etc/shadow access",01/16-19:55:25.407046
,10.101.5.104,10.101.5.83,80
    "(http_inspect) DOUBLE DECODING ATTACK",01/16-19:55:31.595611
,10.101.5.104,10.101.5.83,80
    "WEB-ATTACKS wget command attempt",01/16-19:55:42.057509
,10.101.5.104,10.101.5.83,80
```

c) Ataque ao servidor FTP

Para o ataque ao servidor FTP usamos o xhydra que nos permite fazer ataques de força-bruta e ataques de dicionário.

No Xhydra indicamos o user "root", sendo que é comum existir este utilizador e indicamos um ficheiro de texto com várias palavras-chave para serem testadas.

A máquina atacante tem como IP o 10.101.5.200 e a vitima a máquina com sistema operativo Ubuntu 9.04 e com IP o 10.101.5.83.

Como resultado deste ataque recebemos via correio electrónico vários alertas deste tipo:

```
   Jan 16 22:35:12 n00b-desktop vsftpd:
pam_listfile(vsftpd:auth): Refused user root for service
vsftpd
```

Ao verificar a interface gráfica podemos confirmar a tentativa de intrusão.

Src IP: 10.101.5.200
Multiple failed logins in a small period of time.

Jan 16 22:35:36 n00b-desktop vsftpd: pam_unix(vsftpd:auth): **authentication failure**; logname= uid=0 euid=0 tty=ftp ruser=root rhost=10.101.5.200 user=root
Jan 16 22:35:34 n00b-desktop vsftpd: pam_unix(vsftpd:auth): **authentication failure**; logname= uid=0 euid=0 tty=ftp ruser=root rhost=10.101.5.200 user=root
Jan 16 22:35:30 n00b-desktop vsftpd: pam_unix(vsftpd:auth): **authentication failure**; logname= uid=0 euid=0 tty=ftp ruser=root rhost=10.101.5.200 user=root
Jan 16 22:35:27 n00b-desktop vsftpd: pam_unix(vsftpd:auth): **authentication failure**; logname= uid=0 euid=0 tty=ftp ruser=root rhost=10.101.5.200 user=root
Jan 16 22:35:23 n00b-desktop vsftpd: pam_unix(vsftpd:auth): **authentication failure**; logname= uid=0 euid=0 tty=ftp ruser=root rhost=10.101.5.200 user=root
Jan 16 22:35:19 n00b-desktop vsftpd: pam_unix(vsftpd:auth): **authentication failure**; logname= uid=0 euid=0 tty=ftp ruser=root rhost=10.101.5.200 user=root
Jan 16 22:35:16 n00b-desktop vsftpd: pam_unix(vsftpd:auth): **authentication failure**; logname= uid=0 euid=0 tty=ftp ruser=root rhost=10.101.5.200 user=root

d) Ataque ao servidor de SSH

Para testar a protecção contra ataques a servidores SSH usou-se novamente o xhydra.

No xhydra indicamos o user "root" e indicamos um ficheiro de texto com várias palavras-chave para serem testadas.

Obtivemos as seguintes notificações:

```
   Received From: (ubuntu) 10.101.5.83->/var/log/auth.log
Rule: 5720 fired (level 10) -> "Multiple SSHD authentication failures."
Portion of the log(s):
Jan 11 22:52:54 n00b-desktop sshd[10283]: Failed password for root from
10.101.5.113 port 36863 ssh2
Jan 11 22:52:54 n00b-desktop sshd[10284]: Failed password for root from
10.101.5.113 port 36864 ssh2
Jan 11 22:52:52 n00b-desktop sshd[10283]: Failed password for root from
10.101.5.113 port 36863 ssh2
Jan 11 22:52:52 n00b-desktop sshd[10284]: Failed password for root from
10.101.5.113 port 36864 ssh2
Jan 11 22:52:50 n00b-desktop sshd[10283]: Failed password for root from
10.101.5.113 port 36863 ssh2
Jan 11 22:52:50 n00b-desktop sshd[10284]: Failed password for root from
10.101.5.113 port 36864 ssh2
```

e) Detecção de RootKit

Quando se fez a instalação do agente na máquina onde está a correr o Moodle o OSSEC foi capaz de detectar que existe uma vulnerabilidade no sistema.

Obtivemos a seguinte notificação:

```
   OSSEC HIDS Notification.
2010 Sep 28 21:16:48

Received From: (moodle) 10.101.247.20->rootcheck
Rule: 510 fired (level 7) -> "Host-based anomaly detection event
(rootcheck)."
Portion of the log(s):

Port '55611'(tcp) hidden. Kernel-level rootkit or trojaned version
of netstat.
```

f) Alertas de instalação de programas

Para verificarmos a monitorização local do que acontece nas máquinas instalou-se um programa, neste caso o Microsoft Office 2007 na máquina com Windows Server 2003 e obtivemos esta notificação.

```
   OSSEC HIDS Notification.
2010 Sep 23 16:49:30

Received From: (win2003srv) 10.101.5.85->WinEvtLog
Rule: 18147 fired (level 5) -> "Application Installed."
Portion of the log(s):

WinEvtLog: Application: INFORMATION(11707): MsiInstaller:
Administrator: XTANKI: WINSERVER: Produto: Microsoft Office
Professional Plus 2007 -- Operação de instalação concluída com
êxito.

 --END OF NOTIFICATION
```

6.4.3 Testes de Impacto

Estes testes visam determinar qual o impacto que os agentes têm nas máquinas onde são instalados.

Para verificar este impacto analisou-se o valor de dois parâmetros que reflectem o normal funcionamento de uma máquina.

Os parâmetros foram os seguintes:

- Percentagem de processador utilizado pelo agente.
- Memória RAM utilizada pelo agente.

O método de teste foi o seguinte:

- Sem o agente em execução, medimos a percentagem média de utilização do processador e o consumo de memória, durante 1 minuto e 40 segundos.
- Com o agente em execução, medimos a percentagem média de utilização do processador e o consumo de memória, durante 1 minuto e 40 segundos.

Não foi utilizado um valor "instantâneo" de nenhum dos parâmetros visto que naquele momento poderia ocorrer um pico de consumo de recursos o que levaria a erros durante os testes.

Foi necessário testar o impacto dos agentes em ambiente Windows e ambiente Linux e para tal utilizou-se as seguintes máquinas virtuais:

- AMD Phenom II 2.7 Ghz , com 256mb de ram e com Windows Xp Sp2.
- AMD Phenom II 2.7 Ghz , com 256mb de ram e com Ubuntu 9.10.

a) Windows XP Sp2

Na máquina com sistema operativo Windows foi utilizada a ferramenta de análise de desempenho que vem integrada no conjunto de ferramentas administrativas do Windows Xp, que nos permite a análise da utilização da memória e processador.

Obtiveram-se os seguintes resultados:

- Percentagem de uso do processador durante 1 minuto 40 segundos, sem agente.

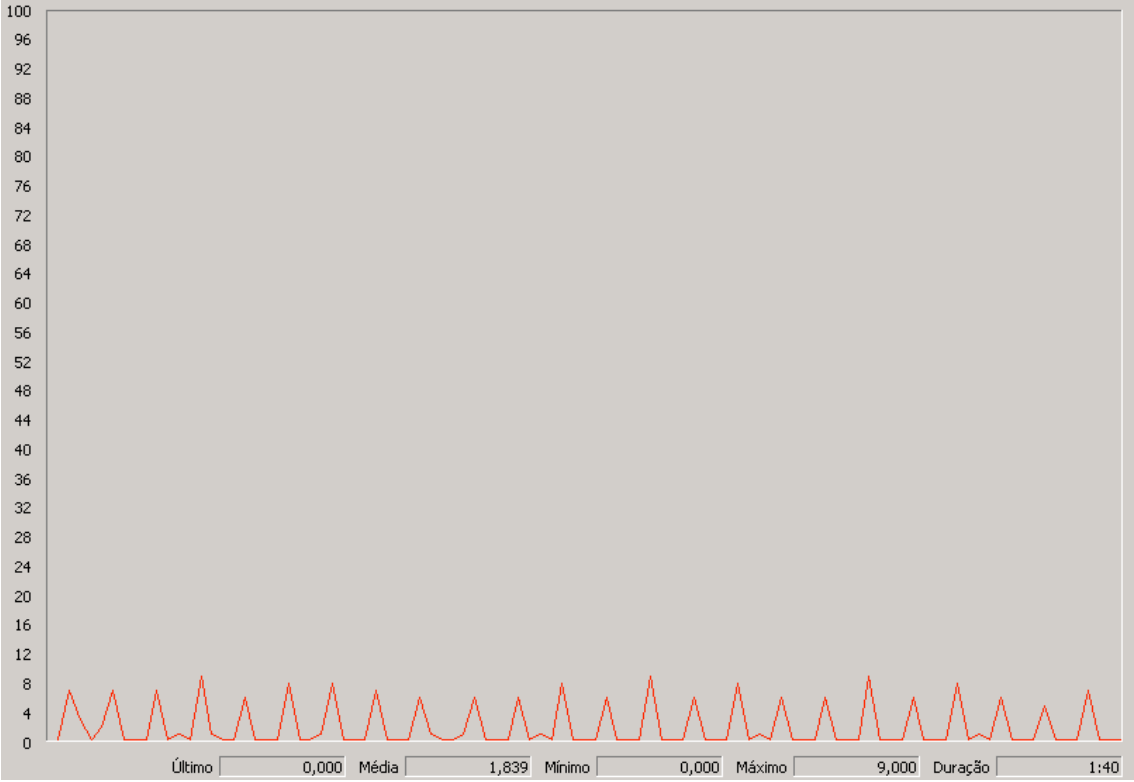

Figura 6.1: Gráfico de utilização de processador, sem agente.

- Percentagem de uso do processador durante 1 minuto 40 segundos, com agente.

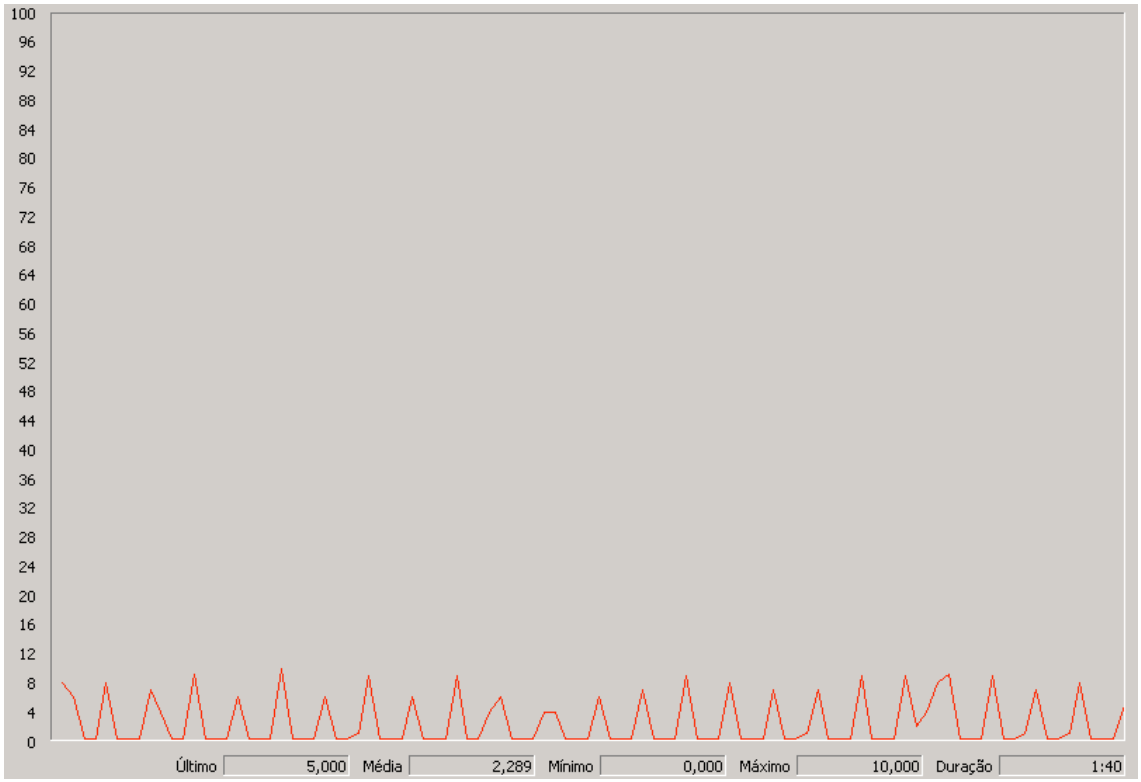

Figura 6.2: Gráfico de utilização de processador, com agente.

- Memória utilizada, durante 1 minuto 40 segundos, sem agente.

Figura 6.3: Gráfico de consumo de memória, sem agente.

- Memória utilizada, durante 1 minuto 40 segundos, com agente.

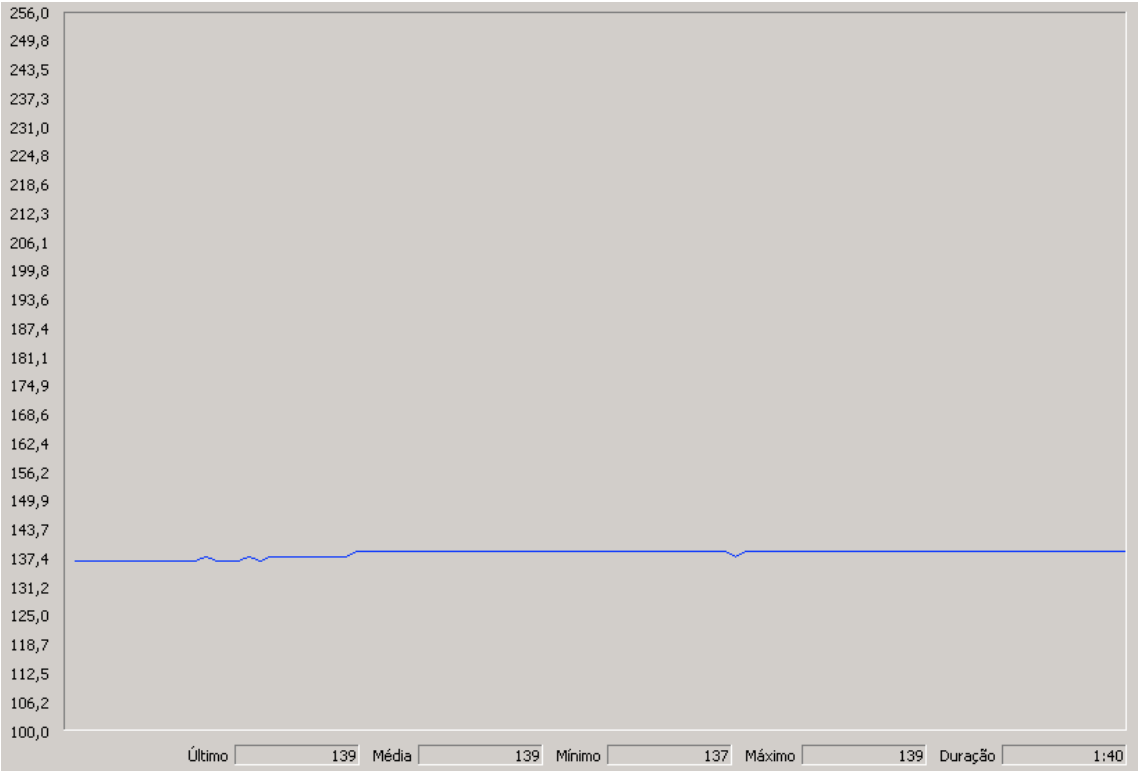

Figura 6.4: Gráfico de consumo de memória, com agente.

b) Ubuntu 9.10

Na máquina com sistema operativo Linux foi utilizada uma ferramenta chamada *System monitor* ferramenta de análise de recursos do sistema, como a percentagem de utilização do processador, a quantidade de memória utilizada ou a lista de processo em execução.

- Percentagem de uso do processador durante 1 minuto 40 segundos, sem agente.

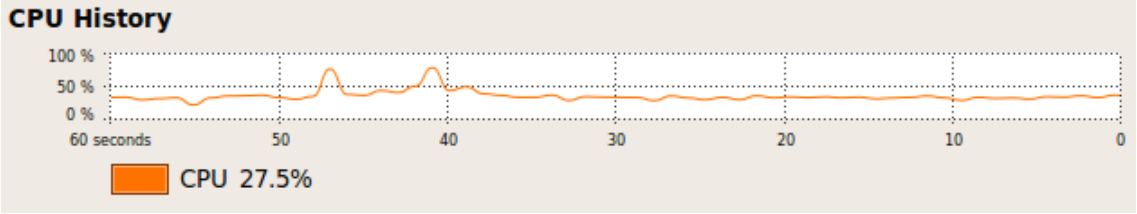

Figura 6.5: Gráfico de utilização de processador, sem agente.

- Percentagem de uso do processador durante 1 minuto 40 segundos, com agente.

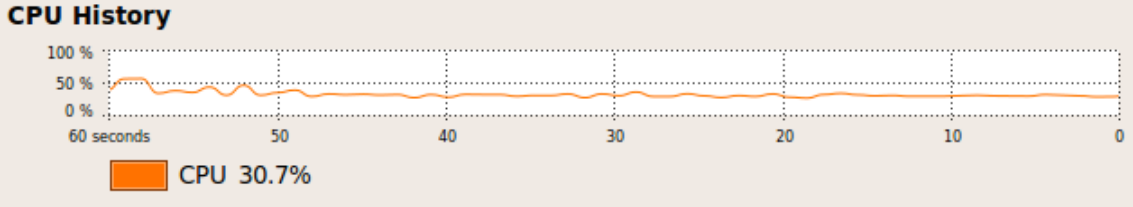

Figura 6.6: Gráfico de utilização de processador, com agente

- Memória utilizada, durante 1 minuto 40 segundos, sem agente.

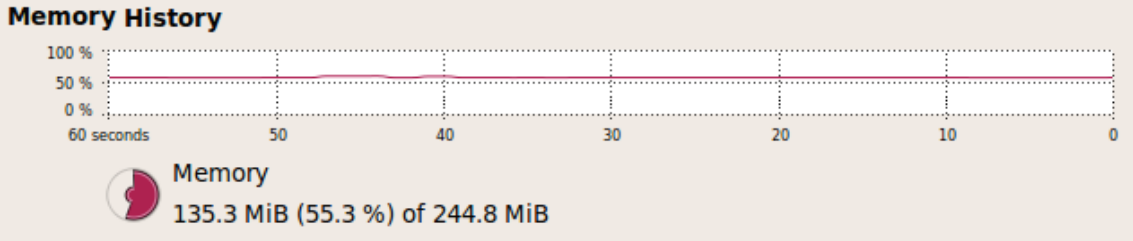

Figura 6.7: Gráfico de consumo de memória, sem agente

- Memória utilizada, durante 1 minuto 40 segundos, com agente.

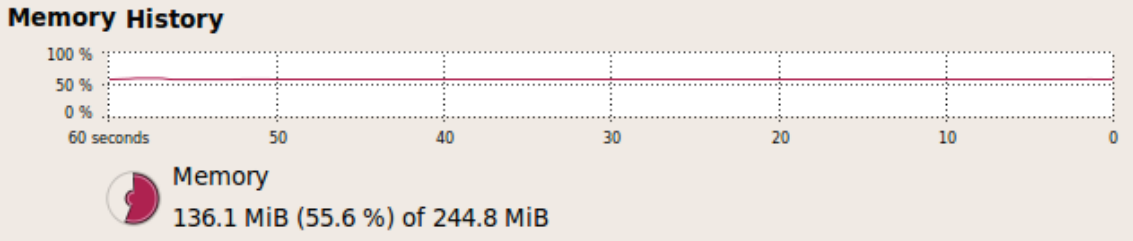

Figura 6.8: Gráfico de consumo de memória, com agente

6.4.4 Tabela de resultados e Conclusão

De seguida são apresentadas as tabelas com os resultados aos testes feitos.

a) Resultados Testes Funcionais

Através dos testes funcionais conseguimos perceber de que forma é que o sistema se comporta em diferentes situações e com diferentes sistemas.

Aqui fica a tabela com os resultados e a informação obtida:

	Ip do Atacante	Detalhe sobre o ataque
Port Scan	sim	sim
Ataque ao servidor Web	sim	sim
Ataque ao servidor FTP	sim	sim
Ataque ao servidor de SSH	sim	sim
Detecção de RootKit	sim	sim
Alertas de instalação de programas	n.a.	n.a.

Tabela 6.1: Tabela de resultados dos testes funcionais

Verificou-se ainda que após a detecção de um possível ataque, a regra de bloqueio de IP funcionou e a máquina atacante ficou sem possibilidade de contacto com a máquina vitima durante 600 segundos.

b) Resultados Testes Impacto

Após os testes os dados recolhidos foram:

- Windows XP Sp2

Windows XP Sp2	Sem Agente	Com agente
Precentagem de Processador utilizado	1,80%	2,30%
Quantidade de memória utilizada	137 MB	139 MB

Tabela 6.2: Tabela de resultados do Windows XP sp2

- Ubuntu 9.10

Ubuntu 9.10	Sem Agente	Com agente
Precentagem de Processador utilizado	27,50%	30,70%
Quantidade de memória utilizada	135,3 MB	136,10 MB

Tabela 6.3: Tabela de resultados do Ubuntu 9.10

Verificou-se que:

- No sistema Windows Xp Sp2 o aumento de utilização do processador foi de 0,5% e a memória utilizada pelo agente foi de 2 MB.
- No sistema Ubuntu 9.10 o aumento de utilização do processador foi de 3,2% e a memória utilizada pelo agente foi de 0,8 MB.

Podemos concluir que o impacto que os agentes terão nas máquinas será mínimo e que não irá influenciar o seu normal funcionamento.

Capítulo 7 Conclusão e Trabalho Futuro

Este trabalho contribuiu para melhorar a segurança da infra-estrutura gerida pelo Centro de Informática. A Firewall existente já faz a filtragem de todo o tráfego fora da zona confiável da rede, mas não é capaz de proteger contra ataques feitos às páginas de internet (como falhas no código ou *SQL Injection)* ou ainda contra serviços para os quais exista um *exploit* disponível. Desta forma será mais fácil detectar e prevenir ataques e proteger os sistemas. Foi possível concluir com sucesso todas as tarefas planeadas para a implementação do sistema. Tarefas estas que incluíram:

- Análise do estado da arte, das várias arquitecturas e as soluções existentes.
- Recolha de informação sobre a infra-estrutura gerida pelo Centro de Informática.
- Análise e enquadramento dos requisitos necessários á implementação do sistema.
- Análise à estrutura e funcionamento da solução escolhida.
- Implementação e configuração da solução escolhida.
- Realização de testes funcionais.

Podemos concluir que para termos um sistema seguro não é suficiente ter uma boa firewall ou sistema de detecção de intrusões. É necessário também ter uma boa equipa de administração de sistemas, políticas de segurança bem definidas e acima de tudo é necessário educar as pessoas para que estas evitem ter comportamentos de risco que possam comprometer as infra-estruturas.

7.1 Dificuldades encontradas

As maiores dificuldades foram encontradas no decorrer dos testes, visto que não foi possível fazer os testes em servidores que já se encontravam em produção. A forma de ultrapassar estas dificuldades foi a criação de máquinas virtuais com uma réplica dos serviços que se encontravam nos servidores originais. Não foi possível também simular ataques para todas a situações em que o sistema cria alertas, por isso optou-se por recriar aqueles mais comuns. Devido à falta de tempo não foi possível testar os ataques nas várias zonas da rede, tais como o Espaço Estudante ou através da rede Wireless (*eduroam*). Os testes foram realizados dentro da rede do Centro de Informática, visto que existe menos influência da Firewall nos testes.

7.2 Trabalhos Futuros

Como trabalho futuro podemos estender o uso do sistema IDS a outras plataformas e situações, como por exemplo:

- Instalação de sensores nos computadores dos laboratórios dos alunos para desta forma ser possível monitorizar falhas ou ataques que possam estar a ocorrer nessas máquinas.
- Caso existam várias tentativas falhadas de login num curto espaço de tempo o sistema deve ser capaz de bloquear a conta e de notificar um possível ataque.
- Automatização de tarefas de registo de aplicações para que seja possível adicionar uma nova chave de autenticação há aplicações que tenham a sua chave expirada.
- Como forma de poupar energia e reduzir custos à instituição, a partir de uma certa hora, desligar todos os computadores que não se encontrem em uso.

Capítulo 8 Referências

[1] William Stallings. Cryptography *and Network Security - Principles and Practice*- Prentice Hall, 2006.

[2] André Zúquete. *Segurança em Redes Informáticas.* FCA - Editora de ·Informática, 2006.

[3] Paulo Sousa, Miguel Correia. *Segurança de Software.* FCA - Editora de Informática, 2010

[4] Paulo Veríssimo, Luís Rodrigues - *Distributed Systems for System Architects*. Kluwer Academic Publishers, 2001

[5] Global Reports of the Symantec Intelligence Quarterly (April – June 2010)

http://www.symantec.com/content/en/us/enterprise/white_papers/b-symc_intelligence_quarterly_apr-jun_2010_21072009.en-us.pdf

[6] Andrew Hay, Daniel Cid, Rory Bray. *OSSEC HIDS Host-Based Intrusion Detection Guide.* Syngress Publishing, Inc., 2008.

[7] Tecnologias de Segurança- Sistemas de Detecção de Intrusões

https://coruja.di.fc.ul.pt/mod/resource/view.php?id=2297

[8] Wikipedia - HoneyPot.

http://en.wikipedia.org/wiki/Honeypot

[9] Sectools - Top 5 Intrusion Detection Tools.
http://sectools.org/ids.html

[10] Ossec - http://www.ossec.net/

[11] Snort - http://www.snort.org/

[12] AntiSniff - http://packetstormsecurity.nl/sniffers/antisniff/

[13] Tripwire - http://www.tripwire.com/

[14] Valhala - http://valhalahoneypot.sourceforge.net/

[15] CISCO - Pix 525 Firewall
http://www.cisco.com/en/US/products/hw/vpndevc/ps2030/ps2118/

[16] Nortel - Alteon 5114 Firewall
http://www.nortel.com/products/01/alteon/asf/asf_5000/collateral/nn110160.pdf

Capítulo 9 Glossário

- **Acess points:** é um equipamento que permite a criação de redes sem fios.
- **Assinaturas digitais:** é um método de autenticação de informação digital tipicamente tratada como análoga à assinatura física em papel.
- **BlowFish:** é uma cifra simétrica com chave de tamanho variável de 32 a 448 bits.
- **Chroot:** é um comando dos sistemas operativos UNIX/Linux, com a intenção de limitar um utilizador a uma directoria, não permitindo que este regrida na árvore de arquivos.
- **Cluster:** é um conjunto de computadores que trabalham para um fim comum.
- **Data Center:** local onde estão concentrados os equipamentos de processamento, armazenamentos de dados e equipamento de rede.
- **Deamon:** é um programa que corre em backgroud.
- **Gateway:** é uma máquina geralmente destinada a ligar redes.
- **Hacker:** é um indivíduo com uma grande aptidão técnica para sistemas computacionais.
- **Hash:** é uma sequência de bits gerada um algoritmo de dispersão capaz de em pouco espaço representar uma grande quantidade de informação.
- **Instant-messaging:** é um software de envio e recepção de mensagens em tempo real.
- **Iptables:** é uma ferramenta que permite a definição de regras de firewall.
- **Keystroke logging:** é a acção de registar as teclas do teclado que são pressionadas.
- **Mac-Adress**: é o endereço físico de uma interface de rede. É um endereço de 48 bits, representado em hexadecimal.
- **Logs**: é um termo utilizador para descrever o processo de registo de eventos relevantes num sistema computacional.

- **Patch:** é um programa de computador que quando executado corrige problemas ou adiciona novas funções a um outro programa de computador.
- **Penetration testing:** são testes que têm como função testar e avaliar os sistemas de seguranças de computadores ou redes.
- **Phishing:** é uma forma de fraude electrónica, caracterizada por simular um site que à partida seria confiável, de forma a puder adquirir informação sensível (nomes, passwords, números de cartões de crédito).
- **Port scanning:** é uma acção que tem como objectivo testar as portas lógicas de um determinado host remoto.
- **Rootkit:** é um programa malicioso, que após instalado permite o acesso privilegiado ao atacante.
- **Routers**: é um equipamento usado para fazer comunicação entre diferentes redes de computadores. A sua principal função é comutar os pacotes recebidos para a porta mais indicada.
- **SNMP Traps** : mensagem enviada por um equipamento de rede para o sistema de gestão da rede.
- **Spam:** são mensagens de correio electrónico com fins publicitários.
- **Spyware:** é um programa de computador que recolhe informação sobre o utilizador e os seus hábitos na internet e envia-os para uma entidade externa, muitas vezes para fins publicitários.
- **Switch**: é um dispositivo utilizado em redes de computadores que permite o envio de tramas entre vários nós.
- **VLAN:** é uma rede local virtual, logicamente independente, podendo coexistir várias no mesmo comutador.
- **Zlib:** é uma biblioteca multi-plataforma de compressão de dados.
- **Zombie/bot:** é um termo usado para computadores utilizados para fazer ataques DOS ou para enviar SPAM.

Capítulo 10 Abreviaturas

ACK	Acknowledge
AD	Active Directory
DHCP	Dynamic Host Configuration Protocol
DNS	Domain Name System
FCUL	Faculdade de Ciências das Universidade de Lisboa
FTP	File Transfer Protocol
FWSM	FireWall Services Module
ICMP	Internet Control Message Protocol
IP	Internet Protocol
MRTG	Multi Router Traffic Grapher
PDF	Portable Document Format
RFID	Radio-Frequency Identification
SSH	Secure Shell
SSO	Single Sign-ON
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
USB	Universal Serial Bus